

# 非线性模型预测控制方法

张友安 王亚锋 刘京茂 著  
盖俊峰 孙玉梅

電子工業出版社·

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书主要介绍作者团队在非线性模型预测控制方面的研究成果。具体包括：第 1 章介绍了预测控制的研究背景与发展及国内外研究现状；第 2、3 章重点研究了预测控制系统的稳定条件；第 4 章设计了三种鲁棒预测控制算法，并比较了几种典型粒子滤波器的估计精度与计算速度；第 5 章针对跟踪问题，设计了非线性预测控制器；第 6、7 章将预测控制应用于具体系统；第 8 章对全书工作进行了回顾和总结，并对未来工作进行了展望。

本书可作为高等院校控制相关专业研究生、高年级本科生的教学参考书。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。  
版权所有，侵权必究。

## 图书在版编目（CIP）数据

非线性模型预测控制方法 / 张友安等著. —北京：电子工业出版社，2017.12

ISBN 978-7-121-32931-9

I. ①非… II. ①张… III. ①非线性—线性模型—预测控制 IV. ①TP273

中国版本图书馆 CIP 数据核字（2017）第 260514 号

策划编辑：朱雨萌

责任编辑：杨秋奎

特约编辑：彭 瑛 赵海军等

印 刷：

装 订：

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱

邮编：100036

开 本：720×1000 1/16 印张：13.75 字数：264 千字

版 次：2017 年 12 月第 1 版

印 次：2017 年 12 月第 1 次印刷

定 价：49.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 [zltz@phei.com.cn](mailto:zltz@phei.com.cn)，盗版侵权举报请发邮件到 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

本书咨询联系方式：（010）88254750。

# 前言

模型预测控制（Model Predictive Control, MPC）是一种在工业过程控制领域得到成功应用的控制策略。由于直接产生于实际工业过程控制，且全面考虑了工业实际需求，具有较高的综合控制质量，因此 MPC 自其诞生之日起便吸引了众多研究者的目光，并逐渐引起了工业控制界和理论界的重视。迄今为止，模型预测控制已被广泛应用于发电、炼油、冶金、化工、汽车、航空航天等领域。MPC 作为一种新兴的控制理论能得到如此成功应用，其主要原因可总结为以下两点：

一是通过滚动优化策略在线求解局部最优问题。模型预测控制是一种动态优化方法，通过优化窗口的滚动，对模型预测输出值和实际输出值进行比较，再通过反馈校正实现系统输出对参考值的跟踪。由于 MPC 采用有限时域优化窗口，仅需求解局部最优问题，大大减少了优化计算量。同时，采用反馈校正可解决多种不确定性问题，包括系统干扰、模型失配等。

二是直接处理约束条件的能力。模型预测控制通过在线优化求解控制律，在其在线优化过程中可以全面考虑各种约束条件（如状态约束、输入约束、输入增量约束、输出约束等），从而得出既能使优化指标最优又能满足各种约束条件的控制律，这是传统的 PID 控制和经典最优控制理论所无法实现的。

上述两个特点使得 MPC 更接近实际的工业生产控制过程，因此得到了众多工程技术人员和研究者的青睐。

预测控制自诞生至今，在工程实践和理论研究方面都取得了重大进展。但现有的具有代表性的学术研究成果大多是基于线性模型描述的系统，目前成熟的 MPC 商用软件也主要针对线性系统开发，即使在过程工业控制技术中，MPC 技术的应用也只局限在某些过程非线性特征不明显的领域。而实际工业生产过程中遇到的对象系统往往具有明显的非线性特征和不确定特性，因此，著者以同时添加终端代价函数和终端状态约束集的非线性预测控制为对象进行了理论研究。

本书内容包括预测控制的终端状态约束集、终端代价函数、鲁棒预测控制、跟踪问题中的预测控制以及预测控制的实际应用等方面。具体包括：第 1 章介绍了预测控制的研究背景与发展、几种典型的非线性预测控制方法以及关于非线性预测控制方法的国内外研究现状。第 2 章介绍了为保证系统稳定性，预测控制的终端状态约束集需要满足的条件，并提出了两种求取最大终端状态约束集的方法：用外包集序列逼近最大终端状态约束集；用子集序列逼近最大终端状态约束集。第 3 章分析了预测控制的终端代价函数分别与终端状态约束集和预测控制消耗的全局性能指标之间的关系，并给出了一种简单的求取终端代价函数的方法；设计了一种具有可变自调节终端代价函数的 NMPC (Nonlinear Model Predictive Control) 控制方案。第 4 章针对存在有界干扰时的非线性系统，设计了三种鲁棒预测控制算法：以集合包含度为优化指标的鲁棒预测控制算法；基于优化线性反馈控制律的鲁棒预测控制算法；基于优化多项式反馈控制律的鲁棒预测控制算法。另外，针对鲁棒预测控制系统中存在的外来干扰，介绍了现有典型非线性滤波器的适用范围，并比较了几种典型粒子滤波器的估计精度与计算速度。第 5 章将跟踪问题按期望输出的形式分为两种类型：对单个稳定期望输出点的逼近；对期望输出轨迹的跟踪。针对这两种跟踪问题，分别设计了非线性预测控制器，并提出了

一种求取次优滚动控制律的方法。第6章介绍了航天器的末端自主交会技术，并将预测控制应用于航天器的末端交会控制。第7章研究了飞行器的姿态机动问题，提出一种在给定时域内将飞行器姿态驱动到设定姿态的模型预测控制算法，并用仿真验证针对飞行器姿态机动的预测控制算法的有效性。第8章对全书工作进行了回顾和总结，并对未来工作进行了展望。

全书共8章。第2、4、6章由张友安与王亚锋共同编写，第3章的3.6节、第5章的5.5节及第7章由盖俊峰与刘京茂（山东南山国际飞行有限公司高工）共同编写，第3、5章的其余各节由张友安、王亚锋与刘京茂共同编写，第1、8章由张友安与孙玉梅（烟台南山学院教授）共同编写。全书由烟台南山学院教授张友安统稿。

本书的部分内容参考和引用了国内外同行专家、学者的最新研究成果，在此特向他们表示由衷的感谢。本书的出版得到了烟台南山学院、海军航空工程学院与电子工业出版社各级领导和朱雨萌编辑的大力支持，在此一并表示感谢！

由于著者学术水平有限，加之出版时间仓促，书中存在错误与不足之处在所难免，敬请广大读者批评指正。



# 目 录

第 1 章 绪论.....	1
1.1 预测控制的研究背景与发展.....	1
1.2 几种典型的非线性预测控制方法.....	3
1.2.1 添加终端零约束的预测控制 .....	4
1.2.2 添加终端状态集约束的预测控制 .....	5
1.2.3 添加终端代价函数的预测控制 .....	6
1.2.4 同时添加终端状态集约束和终端代价函数的预测控制 .....	8
1.3 国内外研究现状.....	10
1.3.1 关于如何求取终端状态约束集 .....	11
1.3.2 关于如何求取终端代价函数 .....	12
1.3.3 关于鲁棒预测控制器设计 .....	13
1.3.4 关于跟踪问题中的预测控制 .....	14
1.4 结构安排 .....	15
第 2 章 求取预测控制的终端状态约束集.....	17
2.1 引言 .....	17
2.2 终端状态集需要满足的三个基本条件 .....	18

2.3 用外包集序列逼近最大终端状态集.....	20
2.3.1 外包集序列的构造方法.....	21
2.3.2 外包集序列的收敛性分析.....	22
2.3.3 支持向量机.....	23
2.3.4 最大终端状态集的外包估计.....	25
2.3.5 用外包集序列逼近最大终端状态集的计算步骤.....	27
2.3.6 仿真算例.....	29
2.4 用子集序列逼近最大终端状态集.....	32
2.4.1 子集序列的构造方法.....	32
2.4.2 子集序列的收敛性分析.....	34
2.4.3 用子集序列逼近最大终端状态集的计算步骤.....	36
2.4.4 仿真算例.....	38
2.5 本章小结.....	40
<b>第3章 求取预测控制的终端代价函数.....</b>	<b>41</b>
3.1 引言.....	41
3.2 预测控制与最优控制的关系.....	42
3.3 终端代价函数与终端状态集之间的关系.....	46
3.4 终端代价函数与全局性能指标之间的关系.....	47
3.5 求取一个合适的终端代价函数.....	52
3.6 具有可变自调节终端代价函数的 NMPC 方案.....	54
3.6.1 问题描述.....	55
3.6.2 对闭环系统性能的证明.....	57
3.6.3 对自调节规则的说明.....	61
3.6.4 几种自调节规则及其性质.....	63
3.7 仿真算例.....	65
3.8 本章小结.....	69



第 4 章 鲁棒预测控制方法 .....	71
4.1 引言 .....	71
4.2 以集合包含度为优化指标的鲁棒预测控制 .....	72
4.2.1 预测可达集与集合包含度的定义 .....	73
4.2.2 鲁棒预测控制器设计 .....	73
4.2.3 预测可达集与终端状态集的计算方法 .....	75
4.2.4 仿真算例 .....	78
4.3 基于优化线性反馈控制律的鲁棒预测控制 .....	80
4.3.1 输入状态大致稳定的概念及相关结论 .....	81
4.3.2 鲁棒预测控制器设计 .....	82
4.3.3 终端代价函数与终端状态集的求取方法 .....	87
4.3.4 仿真算例 .....	89
4.4 基于优化多项式反馈控制律的鲁棒预测控制 .....	90
4.4.1 相关定义 .....	91
4.4.2 区间算法简介 .....	93
4.4.3 鲁棒预测控制器设计 .....	94
4.4.4 仿真算例 .....	97
4.5 鲁棒预测控制系统中的状态估计研究 .....	99
4.5.1 粒子滤波器简介 .....	101
4.5.2 滤波器性能对比仿真 .....	106
4.5.3 CDPF 在鲁棒预测控制中的应用仿真 .....	108
4.6 本章小结 .....	109
第 5 章 跟踪问题中的预测控制 .....	111
5.1 引言 .....	111
5.2 对单个稳定期望输出点的逼近 .....	112
5.2.1 问题描述 .....	112

5.2.2	稳定性分析 .....	113
5.2.3	终端代价函数的求取 .....	116
5.2.4	终端状态约束集的求取 .....	117
5.2.5	鲁棒预测控制器设计 .....	118
5.2.6	仿真算例 .....	119
5.3	对期望输出轨迹的跟踪 .....	120
5.3.1	问题描述 .....	120
5.3.2	稳定性分析 .....	121
5.3.3	预测控制器设计 .....	123
5.3.4	鲁棒预测控制器设计 .....	126
5.3.5	仿真算例 .....	127
5.4	滚动控制律的次优解法 .....	132
5.4.1	基于斯特林插值公式的近似处理 .....	133
5.4.2	近似滚动控制序列的求取 .....	133
5.4.3	预测输出的求取 .....	136
5.4.4	次优控制律的求取 .....	138
5.4.5	仿真算例 .....	140
5.5	基于线性近似和神经网络逼近的非线性系统预测控制 .....	143
5.5.1	问题描述 .....	144
5.5.2	基于线性近似的非线性模型预测控制 .....	145
5.5.3	基于 RBF 神经网络逼近的模型预测控制 .....	154
5.6	本章小结 .....	160
第 6 章	预测控制在航天器末端自主交会中的应用 .....	161
6.1	引言 .....	161
6.2	基于视线坐标系的相对运动方程 .....	162
6.3	应用仿真 .....	168
6.4	本章小结 .....	174

第 7 章 模型预测控制在飞行器姿态机动中的应用.....	175
7.1 引言 .....	175
7.2 飞行器姿态运动方程的转换与解耦.....	176
7.3 模型预测控制器设计.....	181
7.4 应用仿真分析.....	184
7.5 本章小结 .....	187
第 8 章 结论与展望 .....	188
8.1 主要工作总结.....	188
8.2 存在的问题与下一步的研究方向.....	190
参考文献 .....	192



# 第 1 章

## 绪 论

### 1.1 预测控制的研究背景与发展

预测控制从基本的工业控制中发展起来，如发电厂控制、炼油控制等。发展至今，预测控制已成功应用到化工、食品制造、汽车、航空航天、冶金、造纸等领域。预测控制之所以能取得如此大的成功应用，可以归纳为两方面的原因。

(1) 采用滚动优化策略。预测控制将系统在未来一个时域内的动态行为作为优化指标，利用在线优化求取控制量。这意味着在设计预测控制器时，一些干扰因素对系统的影响也可以被提前预测出来。因此，在设计控制器时可以充分考虑这些干扰因素的影响，使得系统输出可以更接近设定值。

(2) 在线优化中充分考虑各种约束条件。预测控制是通过在线优化求解控制律的，在预测控制的在线优化中，可以将各种约束，如输入约束、状态约束、输出约束等纳入优化的约束条件。因此，通过在线优化得到的控制律是满足这些约束条件且使得优化指标最佳的控制律。

尽管预测控制取得了大量的成功应用，但是其仍主要采用线性模型来描述系统。然而，实际工业过程往往都存在固有的非线性环节，采用线性模型来描述系统会产生一定偏差，造成控制精度不高，有时甚至会造成系统不稳定。因此，采用非线性模型来描述系统，进而设计非线性预测控制器成为近年来一个很有吸引力的研究领域。

预测控制，因为其处理带约束系统（包括输入约束、状态约束、输出约束等）的能力而在近年来取得了巨大的发展。其发展历史可以归纳为两个阶段：第一阶段，经典预测控制；第二阶段，现代预测控制。

第一阶段，基于线性模型的经典预测控制。为了适应复杂的工业过程，1978年，Richalet等提出了基于脉冲响应模型的模型预测启发式控制（Model Predictive Heuristic Control, MPHC）或模型算法控制（Model Algorithmic Control, MAC）<sup>[1,2]</sup>。Richalet的方法在工业界得到了很好的应用，也奠定了预测控制发展的基本思路。1980年，Cutler等提出了基于阶跃响应模型的动态矩阵控制（Dynamic Matrix Control, DMC）<sup>[3]</sup>。1982年，Garica等提出了内模控制（Internal Model Control, IMC）<sup>[4]</sup>。Garica的理论使得人们可以从结构的角度的分析预测控制系统，进而理解预测控制的运行机制。1986年，Kuntze等提出了预测函数控制（Predictive Functional Control, PFC）<sup>[5]</sup>。1987年，Clarke等提出了基于可控自回归积分平均滑动模型（CARIMA）的广义预测控制（Generalized Predictive Control, GPC）<sup>[6,7]</sup>。

这些方法对控制对象的模型要求低，且算法简单、容易实现，在实际工业控制中展现了良好的控制性能。

第二阶段，基于非线性模型是现代预测控制。以非线性模型描述系统进而设计非线性预测控制（Nonlinear Model Predictive Control, NMPC）吸引了众多学者的注意，对这类预测控制而言，李雅普诺夫理论是分析其稳定性的主要工具，我们将这种类型的预测控制统称为现代预测控制。

对于主要采用线性模型系统描述的经典预测控制而言,研究方向主要为对已有预测控制算法进行定量分析,通常采用内模控制理论<sup>[4]</sup>对算法的稳定性进行分析。而对于采用非线性模型系统描述的现代预测控制而言,研究方向主要为设计具有稳定性或其他控制性能的预测控制方法,其稳定性分析的主要工具为李雅普诺夫理论,而评估其控制性能的参照体系为最优控制。从这一点上讲,预测控制可以解释为用有限时域优化逼近最优控制的无限时域优化的一种次优控制算法。除了用有限时域的在线滚动优化取代最优控制的无限时域离线优化,预测控制与最优控制并无其他本质区别。

## 1.2 几种典型的非线性预测控制方法

上面已经提到,最优控制被作为评估预测控制的控制性能的参照体系。对此,首先给出最优控制的优化问题。考虑如下离散系统模型

$$x_{k+1} = f(x_k, u_k) \quad (1-1)$$

其中,  $x_k \in R^n$ ,  $u_k \in R^m$  分别为系统在  $k$  时刻的状态和输入;  $f(\cdot, \cdot)$  为关于  $x_k, u_k$  的连续函数, 满足  $f(0, 0) = 0$ 。系统的状态约束和输入约束分别为  $x_k \in X, u_k \in U$ , 满足  $X$  和  $U$  都是紧的, 且都包含原点。控制目标是将系统状态引导至平衡点(此处的平衡点为原点)。

最优控制在初始点  $x_0$  的优化问题  $P_\infty(x_0)$  可描述为

$$\begin{aligned} \min_{u(i, x_0) \in U} J_\infty(u, x_0) &= \sum_{i=0}^{\infty} q(x(i, x_0), u(i, x_0)) \\ \text{s.t. } x(i+1, x_0) &= f(x(i, x_0), u(i, x_0)) \\ x(i+1, x_0) &\in X, u(i, x_0) \in U \end{aligned} \quad (1-2)$$

其中,  $u = \{u(0, x_0), u(1, x_0), u(2, x_0), \dots, u(+\infty, x_0)\}$  表示控制量序列,  $x(0, x_0) = x_0$ ,  $x_0$  表示优化起始点;  $X$  和  $U$  分别为前面所述的状态约束与输

入约束； $q(x, u)$  为阶段指标函数，其形式为  $q(x, u) = x^T Q x + u^T R u$ ， $Q$  和  $R$  分别为对应维数的正定矩阵。

求解优化问题  $P_\infty(x_0)$  是很难的，尤其是当  $f(\cdot, \cdot)$  为非线性函数的时候， $P_\infty(x_0)$  的求解几乎是不可能的。为了避免这个问题，可用预测控制的有限时域优化（或半无限时域优化<sup>[8,9]</sup>）取代最优控制的无限时域优化。但是，这种逼近需要解决一个关键问题——稳定性。对于最优控制，只要无限时域优化存在可行解即可保证其稳定性；对于预测控制，则需要在其在线优化中附加条件以保证其闭环稳定性，常见方法有 4 种。

### 1.2.1 添加终端零约束的预测控制

这类预测控制的代表文献主要有文献[10-13]。其优化问题  $P_N(x_0)$  可描述为

$$\begin{aligned} \min_{u(i, x_0) \in U} J(u, x_0) &= \sum_{i=0}^{N-1} q(x(i, x_0), u(i, x_0)) \\ \text{s.t. } x(i+1, x_0) &= f(x(i, x_0), u(i, x_0)) \\ x(i+1, x_0) &\in X, u(i, x_0) \in U, i = 0, \dots, N-1 \\ x(N, x_0) &= 0 \end{aligned} \quad (1-3)$$

其中， $x(0, x_0) = x_0$ ， $x_0$  表示优化起始点； $N$  为预测步长。

所有满足优化问题式（1-3）有解的状态点的集合称为此类预测控制的吸引域。定义  $J(u, x_0)$  的极小值为  $J^*(x_0)$ ，优化问题式（1-3）的最优输入序列为

$$u^*(x_0) = \{u^*(0, x_0), \dots, u^*(N-1, x_0)\} \quad (1-4)$$

优化问题式（1-3）的最优状态预测轨迹为

$$x^*(x_0) = \{x^*(1, x_0), \dots, x^*(N, x_0)\} \quad (1-5)$$



在实际控制中, 只将  $u^*(0, x_0)$  作用于实际系统, 下一时刻的输入由下一时刻的优化给出, 每一时刻如此反复, 即所谓的预测控制滚动优化, 借此滚动优化可得到滚动控制律

$$\mathbf{u}_{\text{RH}} = \{u^*(0, x_0), u^*(0, x_1), \dots\} \quad (1-6)$$

其中

$$x_{i+1} = f(x_i, u(0, x_i))$$

通过对末端状态强制施加零约束

$$x(N, x_0) = 0 \quad (1-7)$$

可保证在  $x_1 = f(x_0, u(0, x_0))$  时刻的优化问题  $P_N(x_1)$  存在可行控制律

$$\mathbf{u}(x_1) = \{u^*(1, x_0), u^*(2, x_0), \dots, u^*(N-1, x_0), 0\} \quad (1-8)$$

且与此可行控制律对应的性能指标  $J(\mathbf{u}(x_1), x_1)$  满足

$$J(\mathbf{u}(x_1), x_1) - J^*(x_0) = -q(x_0, u^*(0, x_0)) < 0 \quad (1-9)$$

所以有

$$J^*(x_1) - J^*(x_0) \leq J(\mathbf{u}(x_1), x_1) - J^*(x_0) < 0 \quad (1-10)$$

可见  $J^*(\cdot)$  为李雅普诺夫函数, 因此, 系统状态可由滚动控制律  $\mathbf{u}_{\text{RH}}$  引导至原点, 即当采用滚动控制律  $\mathbf{u}_{\text{RH}}$  时, 系统是稳定的。

## 1.2.2 添加终端状态集约束的预测控制

这类预测控制的代表文献有文献[14,15]。其优化问题可描述为

$$\begin{aligned}
 \min_{u(i, x_0) \in U} J(u, x_0) &= \sum_{i=0}^{N-1} q(x(i, x_0), u(i, x_0)) \\
 \text{s.t. } x(i+1, x_0) &= f(x(i, x_0), u(i, x_0)) \\
 x(i+1, x_0) &\in X, u(i, x_0) \in U, i=0, \dots, N-1 \\
 x(N, x_0) &\in X_f
 \end{aligned} \tag{1-11}$$

其中,  $X_f$  表示终端状态约束集, 也称终端状态集或终端约束集, 其为闭集且包含原点, 即  $0 \in X_f$ 。

类似地, 所有满足优化问题式 (1-11) 有解的状态点的集合称为此类预测控制的吸引域。这类预测控制一般采用双模控制结构, 当状态在  $X_f$  外时, 采用滚动控制律  $u_{RH}$ ; 当状态进入  $X_f$  内部时, 采用线性反馈控制律  $u = K_{loc}x \in U$ , 其中,  $K_{loc}$  为线性反馈增益, 这里, 下标 loc 是 local 的缩写, 表示局部的意思。为保证稳定性,  $X_f$  需要满足三个条件。

**条件 1:**  $X_f$  为包含原点且包含于  $X$  的闭集。

**条件 2:**  $X_f$  内的所有点都可由线性反馈控制律  $u = K_{loc}x \in U$  引导至原点。

**条件 3:**  $X_f$  为线性反馈控制不变集, 即对  $\forall x \in X_f$ , 都有  $f(x, K_{loc}x) \in X_f$ 。

### 1.2.3 添加终端代价函数的预测控制

这类预测控制的代表文献有文献[16,17]。其优化问题可描述为

$$\begin{aligned}
 \min_{u(i, x_0) \in U} J(u, x_0) &= \sum_{i=0}^{N-1} q(x(i, x_0), u(i, x_0)) + F(x(N, x_0)) \\
 \text{s.t. } x(i+1, x_0) &= f(x(i, x_0), u(i, x_0)) \\
 x(i+1, x_0) &\in X, u(i, x_0) \in U, i=0, \dots, N-1
 \end{aligned} \tag{1-12}$$

其中,  $F(\cdot)$  为终端代价函数, 满足  $F(0) = 0$ ,  $F(x) \geq \alpha(\|x\|)$ ,  $\alpha(\cdot)$  是  $\mathcal{K}$  类函数 [ $\alpha: R_+ \mapsto R_+$  ( $R_+$  为正实数) 连续且严格递增, 且满足  $\alpha(0) = 0$  ]。

对于这类预测控制来说, 其吸引域定义为所有满足优化问题式 (1-12) 有解且其优化末端状态  $x(N, x_0)$  会自动进入某一原点邻域  $\Omega$  的状态点的集合。另外, 此吸引域必须满足不变集特性, 即对吸引域内的任意点  $x_0$ , 其优化末端状态  $x(N, x_0)$  会自动进入  $\Omega$ , 下一时刻状态  $x_1$  的优化末端状态  $x(N, x_1)$  也会自动进入  $\Omega$ 。为保证稳定性,  $\Omega$  需要满足一个条件。

对  $\forall x \in \Omega$ , 都  $\exists u_{\text{loc}} \in U$ , 使得下式成立

$$F(x) \geq q(x, u_{\text{loc}}) + F(f(x, u_{\text{loc}})) \quad (1-13)$$

且有

$$f(x, u_{\text{loc}}) \in \Omega \quad (1-14)$$

与添加终端零约束的预测控制类似, 定义  $J(u, x_0)$  的极小值为  $J^*(x_0)$ , 最优输入序列为

$$u^*(x_0) = \{u^*(0, x_0), \dots, u^*(N-1, x_0)\}$$

最优状态预测轨迹为

$$x^*(x_0) = \{x^*(1, x_0), \dots, x^*(N, x_0)\}$$

若优化问题式 (1-12) 有解且优化末端状态自动进入  $\Omega$ , 即  $x^*(N, x_0) \in \Omega$ , 则当系统过渡到下一个状态  $x_1$  时, 取可行控制律为

$$u(x_1) = \{u^*(1, x_0), u^*(2, x_0), \dots, u^*(N-1, x_0), u_{\text{loc}}\} \quad (1-15)$$

其中,  $u_{\text{loc}}$  为满足

$$F(x^*(N, x_0)) \geq q(x^*(N, x_0), u_{\text{loc}}) + F(f(x^*(N, x_0), u_{\text{loc}})) \quad (1-16)$$

且

$$f(x^*(N, x_0), u_{\text{loc}}) \in \Omega \quad (1-17)$$

的控制律。

很明显, 与此可行控制律对应的性能指标  $J(u(x_1), x_1)$  满足

$$\begin{aligned}
 & J(u(x_1), x_1) - J^*(x_0) \\
 &= q(x^*(N, x_0), u_{\text{loc}}) + F(f(x^*(N, x_0), u_{\text{loc}})) - \\
 &\quad q(x_0, u^*(0, x_0)) - F(x^*(N, x_0)) \\
 &\leq -q(x_0, u^*(0, x_0))
 \end{aligned} \tag{1-18}$$

所以有

$$\begin{aligned}
 J^*(x_1) - J^*(x_0) &\leq J(u(x_1), x_1) - J^*(x_0) \\
 &\leq -q(x_0, u^*(0, x_0)) \\
 &< 0
 \end{aligned} \tag{1-19}$$

可见  $J^*(\cdot)$  为李雅普诺夫函数, 因此, 当采用滚动控制律  $u_{\text{RH}}$  时, 系统是稳定的。

## 1.2.4 同时添加终端状态集约束和终端代价函数的预测控制

这类预测控制的代表文献有文献[18-29]。其优化问题  $P_N(x_0)$  可被描述为

$$\begin{aligned}
 \min_{u(i, x_0) \in U} J(u, x_0) &= \sum_{i=0}^{N-1} q(x(i, x_0), u(i, x_0)) + F(x(N, x_0)) \\
 \text{s.t. } x(i+1, x_0) &= f(x(i, x_0), u(i, x_0)) \\
 x(i+1, x_0) &\in X, u(i, x_0) \in U, i = 0, \dots, N-1 \\
 x(N, x_0) &\in X_f
 \end{aligned} \tag{1-20}$$

与添加终端代价函数的预测控制类似, 终端代价函数  $F(\cdot)$  满足  $F(0) = 0, F(x) \geq \alpha(\|x\|)$ ,  $\alpha(\cdot)$  是  $\mathcal{K}$  类函数。不同的是, 添加终端代价函数的预测控制在线优化的末端状态会自动进入原点邻域  $\Omega$ , 而此类预测控制则是对末端状态强制施加约束

$$x(N, x_0) \in X_f \tag{1-21}$$

此处的  $X_f$  与 1.2.2 节中的  $X_f$  有所不同, 此处的  $X_f$  需要满足三个条件。

**条件 1:**  $X_f$  为包含原点并包含于  $X$  的闭集。

**条件 2:** 对  $\forall x \in X_f$ , 都  $\exists u_{\text{loc}} \in U$ , 使得

$$F(x) \geq q(x, u_{\text{loc}}) + F(f(x, u_{\text{loc}})) \quad (1-22)$$

**条件 3:**  $X_f$  为控制不变集, 即采用条件 2 中的控制作用时, 有

$$f(x, u_{\text{loc}}) \in X_f \quad (1-23)$$

同样, 定义  $J(u, x_0)$  的极小值为  $J^*(x_0)$ , 最优输入序列为

$$u^*(x_0) = \{u^*(0, x_0), \dots, u^*(N-1, x_0)\}$$

最优状态轨迹为

$$x^*(x_0) = \{x^*(1, x_0), \dots, x^*(N, x_0)\}$$

系统过渡到下一个状态  $x_1$  时, 构造

$$u(x_1) = \{u^*(1, x_0), u^*(2, x_0), \dots, u^*(N-1, x_0), u_{\text{loc}}\} \quad (1-24)$$

其中,  $u_{\text{loc}}$  为满足

$$F(x^*(N, x_0)) \geq q(x^*(N, x_0), u_{\text{loc}}) + F(f(x^*(N, x_0), u_{\text{loc}})) \quad (1-25)$$

且

$$f(x^*(N, x_0), u_{\text{loc}}) \in X_f \quad (1-26)$$

的控制律。

很明显,  $u(x_1)$  是系统在状态  $x_1$  时的一组可行控制律。而与  $u(x_1)$  对应的性能指标  $J(u(x_1), x_1)$  满足

$$\begin{aligned}
& J(u(x_1), x_1) - J^*(x_0) \\
& = q(x^*(N, x_0), u_{\text{loc}}) + F(f(x^*(N, x_0), u_{\text{loc}})) - \\
& \quad q(x_0, u^*(0, x_0)) - F(x^*(N, x_0)) \\
& \leq -q(x_0, u^*(0, x_0))
\end{aligned} \tag{1-27}$$

所以有

$$\begin{aligned}
J^*(x_1) - J^*(x_0) & \leq J(u(x_1), x_1) - J^*(x_0) \\
& \leq -q(x_0, u^*(0, x_0)) \\
& < 0
\end{aligned} \tag{1-28}$$

因此, 当采用滚动控制律  $u_{\text{RH}}$  时, 系统是稳定的。

关于现代预测控制的研究主要集中在这 4 种类型上。其中, 第一种预测控制结构简单、通俗易懂, 但是强行施加的末端状态零约束使得存在可行解的状态点比较少, 所以其吸引域也相对较小; 第二种预测控制与第一种预测控制类似, 但是其控制模态的切换为其工程应用带来一定的阻碍; 第三种预测控制的在线优化中没有附加末端状态约束, 所以其优化的计算负担相对较小, 所耗优化时间也相对较短一些, 但是因为要求优化末端状态自动进入原点邻域  $\Omega$ , 所以其吸引域也比较小; 第四种预测控制虽然附加了末端状态约束, 优化计算负担较大, 但是其理论框架易于分析, 稳定性条件也比较容易实现, 吸引域较之于其他几种也大一些, 所以第四种预测控制也是吸引现有研究目光最多的一种。

### 1.3 国内外研究现状

本节主要以第四种预测控制为对象, 介绍近年来的一些研究成果。前面已经提到, 最优控制被认为是评估预测控制的控制性能的参照体系, 只是用预测控制的有限时域优化逼近最优控制的无限时域优化。在稳定性可以得到

保证的前提下,如何提高预测控制的控制性能近年来吸引了相当多的研究目光。预测控制的控制性能主要包含两个方面:吸引域的大小;全局性能指标消耗(采用预测控制时消耗的阶段指标函数从初始时刻到无穷时刻的积分)的多少。对于同时添加终端状态约束集和终端代价函数的预测控制来说,吸引域的大小主要与预测控制的预测步长和终端状态约束集相关,预测步长越长,终端状态约束集越大,预测控制的吸引域也越大,而终端状态约束集的大小又与终端代价函数相关;全局性能指标主要与预测控制的预测步长和终端代价函数相关,预测步长越长,终端代价函数与最优性能函数(采用最优控制的性能指标的最优解)越相似,全局性能指标消耗得越少。

简而言之,预测控制的控制性能主要与三方面的因素相关:预测步长;终端状态约束集;终端代价函数。虽然预测步长越长,吸引域越大,消耗的全局性能指标越少,但是预测步长的加大会增加在线优化的计算负担,增长控制律的求取时间,降低预测控制的时效性。因而,近年来对于预测控制的控制性能的研究主要集中在两个方面:如何使得终端状态约束集更大;如何求取一个合适的终端代价函数。

在实际的工业系统中都存在不确定性外来干扰,而预测控制器本身虽说具有一定的鲁棒性<sup>[30]</sup>,但是当系统的实际模型与其标称模型差别较大时,系统的稳定性得不到保证,设计具有强鲁棒性的预测控制算法吸引了众多的眼球。

我们的介绍主要包括三个方面:如何求取大的终端状态约束集;如何求取合适的终端代价函数;鲁棒预测控制器设计。另外,再大体介绍一下在求取预测控制优化解方面的一些研究成果。

### 1.3.1 关于如何求取终端状态约束集

终端状态约束集对预测控制来说至关重要,正是由于在预测控制的优化

问题中添加了终端状态约束  $x(N, x_0) \in X_f$ ，才使得系统的稳定性得以保证。终端状态约束集的引入不仅使得系统的稳定性分析更加直观，而且通过终端状态约束集的大小也可确定系统的吸引域大小。因此，求取一个可保证预测控制系统稳定的终端状态约束集至关重要。

文献[8]构造了一个椭球集作为预测控制的终端状态约束集，此椭球集内的任意点都可由线性反馈控制律引导至平衡点。换句话说，作者将系统的线性反馈控制可稳定椭球域作为预测控制的终端状态约束集。考虑到系统存在控制约束，文献[31]采用饱和控制律替代了文献[8]的线性反馈控制律，并将系统的饱和控制稳定域作为预测控制的终端状态约束集，此终端状态约束集内的任意点都可由饱和控制律引导至平衡点。文献[32-34]采用了收缩预测控制的策略以保证系统的稳定性，通过构造逐渐减小的终端状态约束集序列，使得系统状态由较大的终端状态约束集逐渐进入较小的终端状态约束集，以此保证系统的稳定性。文献[35]通过在预测控制的优化问题中引入松弛终端约束来增大预测控制的吸引域。与文献[8]类似，文献[36]也将系统的线性反馈控制可稳定域作为预测控制的终端状态约束集，不过作者并没有采用椭球集来描述终端状态约束集，而是用多面体集来描述终端状态约束集，其得到的多面体集相较于椭球集更大一些。文献[37]与文献[8]和文献[36]类似，将系统的线性反馈控制可稳定域作为预测控制的终端状态约束集，作者利用支持向量机将整个状态空间分为两部分，即线性反馈不可稳定域和线性反馈可稳定域，然后将用支持向量机估计得到的线性反馈可稳定域作为预测控制的终端状态约束集。文献[37]没有将终端状态约束集局限于椭球或多面体形式，对一些系统而言，其所得结果要大得多。

### 1.3.2 关于如何求取终端代价函数

如前所述，衡量一种预测控制算法的好坏主要看两方面因素：吸引域的大小；全局性能指标消耗的多少（将初始状态控制至平衡点所消耗的性能指



标总和)。终端代价函数对预测控制的控制性能起着很关键的作用,选取一个好的终端代价函数不仅能增大系统的吸引域,还能降低消耗的全局性能指标。但是,关于终端代价函数与全局性能指标之间的具体关系及如何求取终端代价函数的文献并不多见。文献[38]针对无终端状态约束的预测控制证明了按比例增大终端代价函数也会增大预测控制的吸引域;文献[39]重点分析了无终端状态约束的预测控制的全局性能指标,给出了全局性能指标的上下界;文献[40]设计了一种预测控制算法,其标称消耗的全局性能指标与最优性能指标一致,但其终端代价函数中  $F(x) = C_f$  在本质上是一种终端等式约束,这会给预测控制的在线优化带来很大的困难;文献[21]给出了一种带比例参数的二次型终端代价函数,并指出通过选择合适的比例系数和二次型矩阵可以保证系统渐近稳定,但其并没有讨论如何选择的问题,也没有讨论不同的选择会对预测控制的控制性能产生何种影响;文献[8]针对连续系统给出了一种简单的求取方法,得到了一个二次型终端代价函数;文献[37]采用支持向量机拟合产生了一个终端代价函数,但这种方法不容易给出终端代价函数的显式表达。

### 1.3.3 关于鲁棒预测控制器设计

对于无干扰系统而言,为保证稳定性,普遍方法是在预测控制的优化问题中添加终端代价函数与终端状态约束<sup>[25]</sup>。前面已经提到,预测控制器本身具有一定的鲁棒性<sup>[30]</sup>,但是当系统的实际模型与其标称模型差别较大时,系统的稳定性得不到保证,此时就需要设计具有强鲁棒性的预测控制算法。现有鲁棒预测控制算法大多数可归入三个范畴:基于多胞模型的鲁棒预测控制;基于逼近模型(如神经网络模型、模糊模型等)的鲁棒预测控制;基于扰动模型的鲁棒预测控制。其中,基于扰动模型的鲁棒预测控制因为物理意义直观明了、适用范围大等优点,吸引了众多的眼球。文献[41,42]针对受扰系统分别设计了滚动时域  $H_\infty$  预测控制器,其中文献[42]的优化问题可以方便

地转化为 LMI 问题来求解；文献[43]针对多胞不确定系统设计了基于椭圆不变集的鲁棒预测控制器；文献[44]用多面体不变集替代了文献[43]中的椭圆不变集，扩大了系统的初始可行域；文献[45,46]定义了基于某一控制量的近似预测可达集，设计了基于近似预测可达集的预测控制器；文献[47]针对有界扰动系统给出了一个通用的输入状态大致稳定的 min-max MPC 控制器；文献[48]以文献[47]为基础，先离线计算出一个线性反馈控制增益，将待优化控制律看成一仿射输入控制（线性反馈控制律与修正量的和），再对修正量进行在线优化求解得到最终控制律。还有一些其他的鲁棒预测控制方法，如文献[49-58]，此处不再一一详述。

系统存在扰动即意味着真实的状态信息是无法得到的，此时，需要采用滤波器对系统状态进行估计。现有的非线性滤波器主要有两类：一类是卡尔曼滤波体系下的非线性滤波器，包括扩展卡尔曼滤波器（EKF）<sup>[59,60]</sup>、无迹卡尔曼滤波器（UKF）<sup>[61,62]</sup>和中心差分滤波器（CDF）<sup>[63-66]</sup>，这几种滤波器都要求系统噪声和量测噪声服从高斯分布；另一类是基于蒙特卡罗方法的粒子滤波器（PF）<sup>[67-71]</sup>，与卡尔曼滤波体系下的滤波器不同，粒子滤波器不要求系统噪声和量测噪声服从高斯分布。按照建议分布的不同选取，现有典型粒子滤波器有以下 4 种：直接将预测分布作为建议分布的 PF；用 EKF 来传递建议分布的扩展卡尔曼粒子滤波器（EKPF）<sup>[72]</sup>；用 UKF 来传递建议分布的无迹卡尔曼粒子滤波器（UKPF）<sup>[73]</sup>；用 CDF 来传递建议分布的中心差分粒子滤波器（CDPF）<sup>[74]</sup>。此外，对基于集合理论的集员滤波器（SMF）<sup>[76-81]</sup>及基于模型误差最小准则的预测滤波器<sup>[82-84]</sup>，也有不少学者进行了相关研究。

### 1.3.4 关于跟踪问题中的预测控制

对于很多系统来说，有时要求其系统输出稳定在某一个期望输出点或跟踪某一条期望输出轨迹，这就要求研究跟踪问题中的预测控制器设计。跟踪问题中的预测控制器设计相对镇定问题中的预测控制器设计来说要难得多。

文献[85-90]分别针对跟踪问题中的线性系统设计了预测控制器。其中,文献[85]设计了一个指令规划器使得系统的性能指标达到最小化;文献[86]指出线性约束系统的任意收敛域都可以作为跟踪的收敛域;文献[87]针对线性系统对常值期望输出的跟踪设计了一种预测控制器,其优化问题可以用多参数二次规划进行求解;文献[88]针对线性约束系统对分段常值期望输出的跟踪设计了一种双模预测控制器;文献[89]同样针对线性约束系统对分段常值期望输出的跟踪设计了预测控制器,在期望输出点不稳定时,文献[89]的控制器可将系统输出引导至离此不稳定期望输出点最近的一个稳定输出点;文献[90]通过增大跟踪误差惩罚项的权重减小了最优性的丢失。对于非线性系统在跟踪问题中的预测控制器设计,现有文献并不多。文献[91]设计了一种收缩预测控制器,使得非线性系统的输出可到达某一事先设计好的稳定目标点。另外一些文献主要采用以其他易于设计控制律的模型对原非线性系统进行逼近,在此基础上设计预测控制器,如模糊模型<sup>[92,93]</sup>、神经网络模型、多模型<sup>[94,95]</sup>、支持向量机<sup>[96]</sup>、多项式逼近模型<sup>[97]</sup>、增量型最小化模型<sup>[98]</sup>、灰色模型<sup>[99]</sup>、小波网络模型、线性近似模型<sup>[100-102]</sup>等。

## 1.4 结构安排

本书主要以同时添加终端代价函数和终端状态约束集的预测控制为对象,进行了较深入的研究工作,具体各章内容安排如下。

第1章介绍了本书的研究背景、预测控制的发展、几种典型的非线性预测控制方法及关于非线性预测控制方法的国内外研究现状。

第2章介绍了为保证系统稳定性,预测控制的终端状态约束集需要满足的条件,并提出了两种求取最大终端状态约束集的方法:用外包集序列逼近最大终端状态约束集;用子集序列逼近最大终端状态约束集。

第3章分析了预测控制的终端代价函数分别与终端状态约束集和预测控制消耗的全局性能指标之间的关系,并给出了一种简单的求取终端代价函数的方法;提出了一种具有自调节终端代价函数的模型预测控制方案,使终端代价函数能随着控制过程的推进而进行适当的调整,可用较少的全局性能指标消耗保证闭环系统的性能。

第4章针对存在有界干扰时的非线性系统设计了三种鲁棒预测控制算法:以集合包含度为优化指标的鲁棒预测控制算法;基于优化线性反馈控制律的鲁棒预测控制算法;基于优化多项式反馈控制律的鲁棒预测控制算法。另外,针对鲁棒预测控制系统中存在的外来干扰,介绍了现有典型非线性滤波器的适用范围,并比较了几种典型粒子滤波器的估计精度与计算速度。

第5章将跟踪问题按期望输出的形式分为两种类型:对单个稳定期望输出点的逼近;对期望输出轨迹的跟踪。针对这两种跟踪问题,分别设计了非线性预测控制器,并提出了一种求取次优滚动控制律的方法。另外,给出了一种基于线性近似的非线性模型预测控制方法,并在此基础上提出了一种基于RBF神经网络逼近的非线性模型预测控制方案。

第6章介绍了航天器的末端自主交会技术,并将预测控制应用于航天器的末端交会控制。

第7章研究了飞行器的姿态机动问题,提出一种在给定时域内将飞行器姿态驱动到设定姿态的模型预测控制算法。

第8章对全书工作进行了回顾和总结,并对未来工作进行了展望。

## 第 2 章

# 求取预测控制的终端状态约束集

### 2.1 引言

众所周知，增大预测步长、增大终端状态约束集（终端状态集）能增大预测控制的吸引域。但是，如前所述，预测步长的加大会增加在线优化的计算负担，增长控制律的求取时间，降低预测控制的时效性。因此，以增大终端状态约束集为手段来增大预测控制的吸引域吸引了众多研究者的目光。文献[8]用一个线性反馈控制可稳定椭球域作为预测控制的终端状态约束集；文献[31]用系统的饱和控制稳定域作为预测控制的终端状态约束集；文献[36]用一个线性反馈控制可稳定多面体集作为预测控制的终端状态约束集；文献[37]也将线性反馈控制可稳定域用作终端状态约束集，但是，作者没有将终端状态约束集局限于椭球或多面体形式，而是用支持向量机将线性反馈控制可稳定域从整个状态空间中分离出来，因此，对一些系统而言，其所得结果要大得多。

上述文献所采用的方法本质上都基于同一种思想：事先给定某一控制律（如线性反馈控制律、饱和控制律等），再计算出与此控制律相对应的可稳定

域,并将此稳定域用作预测控制的终端状态集。因此,这些方法相对比较保守。本章同样以带终端状态约束集和终端代价函数的预测控制为研究对象,重点讨论如何求取预测控制的终端状态约束集。

受文献[37]的启发,本章也采用支持向量机将整个状态空间分为两个区域,一个区域内的所有点都满足终端状态集需要满足的条件,另一个区域内的点不满足。对此,首先归纳出为保证闭环系统稳定,终端状态集需要满足的三个基本条件<sup>[25]</sup>;其次,设计了两种迭代逼近的方法,并且分别证明了当迭代至无穷次数时,所得到的状态区域就是最大终端状态集;最后,分别采用支持向量机对每一步迭代计算的过渡状态集进行集合近似估计。

## 2.2 终端状态集需要满足的三个基本条件

为方便起见,重写系统模型如下:

$$x_{k+1} = f(x_k, u_k) \quad (2-1)$$

其中,  $x_k \in R^n, u_k \in R^m$  分别为系统在采样时刻  $k$  的状态和输入;  $f(\cdot, \cdot)$  为关于  $x_k, u_k$  的非线性连续函数,满足  $f(0,0)=0$ 。系统的状态约束和输入约束分别为  $x_k \in X, u_k \in U$ , 满足  $X$  和  $U$  都是紧的,且都包含原点。

带终端状态约束集和终端代价函数的预测控制的优化问题  $P_N(x_k)$  可描述为

$$\begin{aligned} \min_{u(i, x_k) \in U} J(u, x_k) &= \sum_{i=0}^{N-1} q(x(i, x_k), u(i, x_k)) + F(x(N, x_k)) \\ \text{s.t. } x(i+1, x_k) &= f(x(i, x_k), u(i, x_k)) \\ x(i+1, x_k) &\in X, u(i, x_k) \in U, i=0, \dots, N-1 \\ x(N, x_k) &\in X_f \end{aligned} \quad (2-2)$$

其中,  $x(0, x_k) = x_k$ ,  $x_k$  表示优化起始点;  $N$  为预测步长;  $X_f$  为终端状态约束集;  $F(\cdot)$  为终端代价函数, 满足  $F(0) = 0, F(x) \geq \alpha(\|x\|)$ ,  $\alpha(\cdot)$  是  $\mathcal{K}$  类函数。

定义  $J^*(x_k)$  为  $J(u, x_k)$  的极小值,  $u^*(x_k) = \{u^*(0, x_k), \dots, u^*(N-1, x_k)\}$  为其最优输入序列,  $x^*(x_k) = \{x^*(1, x_k), \dots, x^*(N, x_k)\}$  为相应的最优状态轨迹。在实际控制中, 只将  $u^*(0, x_k)$  作用于实际系统, 下一时刻的输入由下一时刻的优化给出, 如此反复, 即可得到滚动控制律  $u_{RH} = \{u^*(0, x_k), u^*(0, x_{k+1}), \dots\}$ 。

在第1章中已经提到, 若终端状态集  $X_f$  满足三个条件, 即可保证预测控制系统的闭环稳定性<sup>[25]</sup>, 如引理2.1所示。

**引理 2.1** 定义  $\Gamma_N := \{x \in X \mid x^*(N, x) \in X_f\}$ , 对任意的  $x \in \Gamma_N$ , 如果  $X_f$  满足三个条件:

(C1)  $X_f$  为闭集, 且  $0 \in X_f \subseteq X$ 。

(C2)  $F(\cdot)$  的李雅普诺夫特性。对任意的  $x \in X_f$ , 都有

$$F(x) \geq \min_{u \in U} \{q(x, u) + F(f(x, u))\}$$

(C3) 不变集特性。对任意的  $x \in X_f$ , 采用条件 (C2) 中的最优控制作用时, 满足

$$f(x, u) \in X_f$$

则采用滚动控制律  $u_{RH}$  可使  $x$  最终收敛到零。

引理2.1由文献[25]整理而来, 具体证明可查阅文献[25]。

$\Gamma_N$  称为预测控制的吸引域, 如前所述, 增大终端状态集  $X_f$  可增大  $\Gamma_N$ 。定义  $X_{f, \max}$  为满足条件 (C1)、(C2)、(C3) 且最大的  $X_f$ , 本章要解决的问题即可描述为: 针对系统式 (2-1), 对于给定的终端代价函数  $F(\cdot)$ , 估计出与其对应的  $X_{f, \max}$ 。

## 2.3 用外包集序列逼近最大终端状态集

至今,已有很多相关文献对  $X_f$  的构造提出了切实可行的方法。但是,如前所述,这些方法很多都是事先给定某一控制律  $u = k(x)$  (如线性反馈控制律、饱和控制律等),再计算出与此控制律相应的终端状态集  $X_f$ ,使得对于任意点  $x \in X_f$ , 满足

$$F(x) \geq q(x, k(x)) + F(f(x, k(x)))$$

且

$$\begin{aligned} k(x) &\in U \\ f(x, k(x)) &\in X_f \end{aligned}$$

这种方法对  $X_f$  的构造比较保守,并没有最大限度地逼近  $X_{f,\max}$ 。本章直接从条件 (C1)、(C2)、(C3) 出发构造  $X_f$ , 策略为: 首先通过条件 (C2)、(C3) 构造  $X_f$ , 然后验证  $X_f$  是否满足条件 (C1)。

**注 2.1**  $X_f$  的构造与  $X_{f,\max}$  的逼近都是在  $X$  的内部进行的,且自始至终都包含了原点,也就是说,  $0 \in X_f \in X$  提前就已满足。因此,此处对  $X_f$  的构造以条件 (C2) 和 (C3) 为基础,待得出结果后再通过证明其为闭集以满足条件 (C1)。

基于条件 (C2) 和 (C3), 可将  $X_f$  定义为

$$X_f := \{x \in X \mid F(x) \geq F_{X_f}^*(x)\} \quad (2-3)$$

其中,  $F_{X_f}^*(x)$  是下述优化问题的极小值:

$$\begin{aligned} \min_{u \in U} F_{X_f}(x) &= q(x, u) + F(f(x, u)) \\ \text{s.t.} \quad &f(x, u) \in X_f \end{aligned} \quad (2-4)$$



显然, 在  $X_f$  未知时, 对某一点  $x \in X$ , 直接依据式 (2-3)、式 (2-4) 是无法判别其是否属于  $X_f$  的, 难点就在于优化问题式 (2-4) 中的状态约束用到了  $X_f$  本身。为避免这个问题, 我们采用以外包集进行迭代逼近的方法对  $X_f$  进行估计, 首先给出一个最初估计  $X_f^0 \supseteq X_f$ , 借由  $X_f^0$  得到一个更精确的估计  $X_f^1$ , 如此反复, 依次得到  $X_f^2, X_f^3, \dots$ , 满足  $X_f^0 \supseteq X_f^1 \supseteq \dots \supseteq X_f^j \supseteq X_f$ 。随着  $j$  的增大,  $X_f^j$  逐渐收敛到某一集合  $X_f^{+\infty}$ , 此  $X_f^{+\infty}$  即  $X_f$  的最终估计。

### 2.3.1 外包集序列的构造方法

用外包集序列逼近最大终端状态集的示意图如图 2-1 所示。

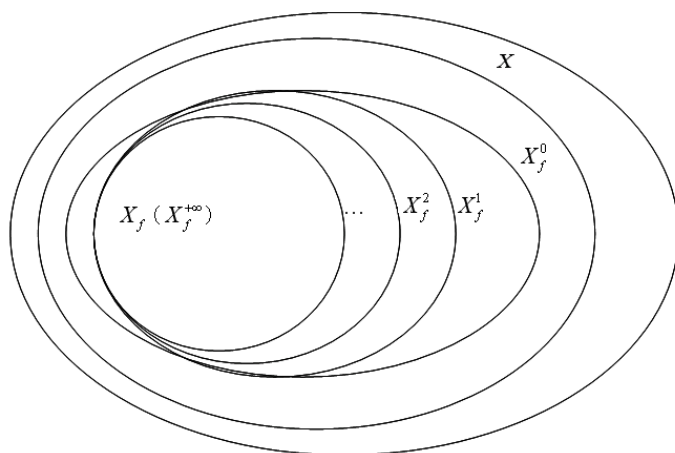


图 2-1 用外包集序列逼近最大终端状态集的示意图

在图 2-1 中,  $X_f^0$  的定义为

$$X_f^0 := \{x \in X \mid F(x) \geq F_{X_f^0}^*(x)\} \quad (2-5)$$

其中,  $F_{X_f^0}^*(x)$  是下述优化问题的极小值:

$$\begin{aligned} \min_{u \in U} F_{X_f^0}(x) &= q(x, u) + F(f(x, u)) \\ \text{s.t.} \quad & f(x, u) \in X \end{aligned} \quad (2-6)$$

类似地,  $X_f^1$  的定义为

$$X_f^1 := \{x \in X \mid F(x) \geq F_{X_f^1}^*(x)\} \quad (2-7)$$

其中,  $F_{X_f^1}^*(x)$  是下述优化问题的极小值:

$$\begin{aligned} \min_{u \in U} F_{X_f^1}(x) &= q(x, u) + F(f(x, u)) \\ \text{s.t.} \quad & f(x, u) \in X_f^0 \end{aligned} \quad (2-8)$$

以此类推,  $X_f^j$  的定义为

$$X_f^j := \{x \in X \mid F(x) \geq F_{X_f^j}^*(x)\} \quad (2-9)$$

其中,  $F_{X_f^j}^*(x)$  是下述优化问题的极小值:

$$\begin{aligned} \min_{u \in U} F_{X_f^j}(x) &= q(x, u) + F(f(x, u)) \\ \text{s.t.} \quad & f(x, u) \in X_f^{j-1} \end{aligned} \quad (2-10)$$

### 2.3.2 外包集序列的收敛性分析

用 2.3.1 节的方法构造的  $X_f^j$  是否收敛? 如果收敛, 又是否收敛于  $X_{f, \max}$  呢? 定理 2.1 为  $X_f^j$  的收敛性提供了答案。

**定理 2.1** 对于式 (2-9) 中定义的  $X_f^j$ , 当  $j \rightarrow +\infty$  时, 有  $X_f^j \rightarrow X_{f, \max}$ 。

证明采用反证法。

(I) 假设存在  $X_{\text{spo}} \supset X_{f, \max}$ , 使得  $j \rightarrow +\infty$  时,  $X_f^j \rightarrow X_{\text{spo}}$ , 则对任意的  $x \in X_{\text{spo}}$ , 都满足

$$F(x) \geq \min_{u \in U} \{q(x, u) + F(f(x, u))\}$$

且

$$f(x, u) \in X_{\text{spo}}$$

这与  $X_{f, \max}$  是满足这两个条件的最大集矛盾。

(II) 假设存在  $X_{\text{spo}} \subset X_{f, \max}$ , 使得  $j \rightarrow +\infty$  时,  $X_f^j \rightarrow X_{\text{spo}}$ , 则存在  $0 \leq N < +\infty$ , 使得  $X_f^N \supseteq X_{f, \max}$ , 且  $X_{f, \max} \setminus X_f^{N+1} \neq \emptyset$ , 其中,  $\emptyset$  意指空集。取任意点  $x \in X_{f, \max} \setminus X_f^{N+1}$ , 显然,  $x$  满足

$$F(x) \geq \min_{u \in U} \{q(x, u) + F(f(x, u))\}$$

且  $f(x, u) \in X_{f, \max} \subseteq X_f^N$ 。又因  $x \in X_f^N$ , 故  $x$  满足  $X_f^{N+1}$  的定义, 即  $x \in X_f^{N+1}$ , 显然矛盾。

证毕。

**注 2.2** 在实际计算过程中, 因时间有限, 是无法计算到  $j \rightarrow +\infty$  的。若迭代到  $j = N_{\text{stop}}$  步时,  $X_f^{N_{\text{stop}}}$  与  $X_f^{N_{\text{stop}}-1}$  很接近 (基本重合), 即可将  $X_f^{N_{\text{stop}}}$  作为对  $X_{f, \max}$  的最终估计。可以看出,  $X_f^{N_{\text{stop}}}$  本质上只是  $X_{f, \max}$  的一个外包集, 将  $X_f^{N_{\text{stop}}}$  作为终端状态集求取预测控制的吸引域可能会包含一些本不属于吸引域的状态点, 这是这种方法的不足之处。

鉴于支持向量机在低样本数情况下也有很好的寻优能力, 我们选用支持向量机作为集合估计工具, 依次估计出  $X_f^0, X_f^1, \dots, X_f^j$ 。具体思路为: 采用支持向量机寻找确定每一个集合的最优分类面, 由此最优分类面从状态空间中依次分离出  $X_f^0, X_f^1, \dots, X_f^j$ 。

### 2.3.3 支持向量机

支持向量机是统计学习理论中最年轻的部分, 其目标是在有限样本的情况下得到一个最优分类面。对于非线性问题, 支持向量机通过非线性变换将其转换到高维的特征空间, 在高维特征空间中构造线性判别函数来实现原空间中的非线性判别函数。

以将  $X$  分成  $A$  与  $X \setminus A$  两类, 即从  $X$  中估计  $A$  为例。从  $X$  中抽取任意点, 并对每一个  $x_i \in A$ , 引入变量  $y_i = +1$ ; 类似地, 对每一个  $x_i \in X \setminus A$ , 引入变量  $y_i = -1$ 。定义  $I^+ := \{i: y_i = +1\}$ ,  $I^- := \{i: y_i = -1\}$ 。

采用支持向量机在  $A$  与  $X \setminus A$  之间找到一个最优分类面

$$O(x) := w \cdot \phi(x) + b = 0 \quad (2-11)$$

满足  $O(x) \geq 0$  的点属于  $A$ , 否则属于  $X \setminus A$ 。而  $O(x)$  可由下面的优化问题求取:

$$\begin{aligned} \min_{\alpha} & \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \phi(x_i) \cdot \phi(x_j) - \sum_i \alpha_i \\ \text{s.t.} & \sum_i \alpha_i y_i = 0 \\ & 0 \leq \alpha_i \leq C, \quad \forall i \in I^+; \quad \alpha_i \geq 0, \quad \forall i \in I^- \end{aligned} \quad (2-12)$$

其中, 内积  $\phi(x_i) \cdot \phi(x_j)$  可由下述高斯核函数代替:

$$\begin{aligned} \phi(x) \phi(x_i) &= \ker(x_i, x) \\ &= \exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) \end{aligned} \quad (2-13)$$

关于支持向量机的研究, 有大量文献可参阅, 如文献[103-111], 但其超出我们的研究范围, 我们不作详细介绍。此处, 以从  $X$  估计出  $X_f^0$  为例, 简单介绍用支持向量机进行集合估计的原理。首先, 通过求解式 (2-6) 描述的非线性优化得到与每一个  $x_i$  对应的  $y_i$ , 并将所有训练样本  $\{x_i, y_i\}$  输入支持向量机; 其次, 支持向量机通过求解式 (2-12), 得到与每一个  $\{x_i, y_i\}$  对应的  $\alpha_i$ ; 最后, 支持向量机由训练样本中选取  $P_0$  个支持向量, 并由此  $P_0$  个支持向量确定估计  $X_f^0$  所需的最优分类面。

$$O^0(x) = \sum_{i=1}^{P_0} w_i \cdot \ker(x_i, x) + b \quad (2-14)$$

其中,  $w_i = \alpha_i y_i$  表示权重系数,  $b = -\sum_i^{P_0} \alpha_i$  表示分类阈值。得到  $O^0(x)$  后,  $X_f^0$

即可由其表示为

$$X_f^0 \triangleq \{x \in X \mid O^0(x) \geq 0\} \quad (2-15)$$

### 2.3.4 最大终端状态集的外包估计

如前所述, 最大终端状态集  $X_{f,\max}$  的估计是一个渐近的逼近过程, 首先估计  $X_f^0$ , 再依次估计  $X_f^1, X_f^2, \dots$ , 当  $X_f^j$  与  $X_f^{j-1}$  很接近时, 即可将  $X_f^j$  作为  $X_{f,\max}$  的最终估计。用支持向量机对这些过渡状态集进行估计首先要解决训练样本的获取问题。

以估计  $X_f^0$  为例, 抽取任意点  $x_i \in X$ ,  $y_i$  的判别如下:

$$\text{如果 } F(x_i) \geq F_{X_f^0}^*(x_i)$$

$$\text{则, } y_i = +1$$

$$\text{否则, } y_i = -1$$

其中,  $F_{X_f^0}^*(x_i)$  是优化问题式 (2-6) 的极值。

得到训练样本后, 将其输入支持向量机, 得到最优分类面  $O^0(x) = 0$ , 最终确定  $X_f^0$ , 即  $X_f^0 \triangleq \{x \in X \mid O^0(x) \geq 0\}$ 。

以  $X_f^0$  为基础, 可对  $X_f^1$  进行估计。类似地, 抽取任意点  $x_i \in X_f^0$ , 判别与其对应的  $y_i$ :

$$\text{如果 } F(x_i) \geq F_{X_f^1}^*(x_i)$$

$$\text{则, } y_i = +1$$

$$\text{否则, } y_i = -1$$

其中,  $F_{X_f^1}^*(x_i)$  是优化问题式 (2-8) 的极值。

同理，将训练样本输入支持向量机可得到最优分类面  $O^1(x)=0$ ，如此反复，可得到一系列的  $O^j(x)=0$ 。

当  $j = N_{\text{stop}}$  时，若满足

$$\sum_{i=1}^{P_{N_{\text{stop}}}} \|O^{N_{\text{stop}}}(x_i) - O^{N_{\text{stop}}-1}(x_i)\| \leq \varepsilon P_{N_{\text{stop}}} \quad (2-16)$$

即可认为  $X_f^{N_{\text{stop}}}$  与  $X_f^{N_{\text{stop}}-1}$  比较接近，并将  $X_f^{N_{\text{stop}}}$  作为对  $X_{f,\max}$  的最终估计。其中， $x_i \in X_{\text{sup},N_{\text{stop}}}$ ， $X_{\text{sup},N_{\text{stop}}}$  表示  $j = N_{\text{stop}}$  时的支持向量集， $P_{N_{\text{stop}}}$  表示  $j = N_{\text{stop}}$  时的支持向量个数， $\varepsilon$  为某一设定的阈值，其值越小， $X_f^{N_{\text{stop}}}$  对  $X_{f,\max}$  的逼近越准确。最终， $X_{f,\max}$  可用其外包集来近似估计： $X_{f,\max} \triangleq \{x \in X \mid O^{N_{\text{stop}}}(x) \geq 0\}$ 。

**注 2.3** 采用式 (2-16) 来判断两个集合比较接近的依据是：状态集本身由其最优分类面确定，而以支持向量机为工具进行分类得到的最优分类面本质上是一个函数表达式，这个函数表达式的参数由其所选取的支持向量确定。换句话说，不同的最优分类面的函数值在其支持向量处是最敏感的。因此，两个集合是否接近可以简单地用两个分类函数在其中一个分类面所选取的支持向量处的差值的范数来判断。所以，式 (2-16) 中用到的支持向量集也可以用  $X_{\text{sup},N_{\text{stop}}-1}$  来替代。

**定理 2.2** 由此方法得到的  $X_{f,\max}$  满足条件 (C1)。

证明：

(I) 根据  $X_f$  的构造与  $X_{f,\max}$  的逼近原理，明显可以看出， $X_{f,\max} \in X$ 。

(II) 取原点状态  $x=0$ 。当  $u=0$  时，有

$$0 \in X \text{ 且 } F(0) = F_0^*(0) = q(0,0) + F(f(0,0)) = 0$$

显然，0 点满足条件 (C2) 和 (C3)，因此， $0 \in X_{f,\max}$ 。

(III) 用支持向量机估计得到的  $X_{f,\max}$  的表达式为

$$X_{f,\max} \triangleq \{x \in X \mid O^{N_{\text{stop}}}(x) \geq 0\}$$

可以看出, 边界包含于此估计集合中, 因此, 此估计集合为闭集。由 (I)、(II)、(III) 可知, 用支持向量机估计得到的  $X_{f,\max}$  满足条件 (C1)。

证毕。

**注 2.4** 整个  $X_f$  的构造与  $X_{f,\max}$  的逼近过程都是以条件 (C2) 和 (C3) 为基础的, 而最后得到的估计集合也满足条件 (C2) 和 (C3), 因此, 用此方法得到的  $X_{f,\max}$  可作为预测控制的终端状态集。

### 2.3.5 用外包集序列逼近最大终端状态集的计算步骤

综上所述, 采用此方法求取预测控制终端状态集的步骤可整理如下。

**第一步:** 设置 SVM 的训练样本数及循环终止阈值  $\varepsilon$ 。

**第二步:** 导入循环  $j=0, 1, 2, \dots$ , 用 SVM 求取最优分类面  $O^j(x)=0$ , 确定  $X_f^j$ :  $X_f^j \triangleq \{x \in X_f^{j-1} \mid O^j(x) \geq 0\}$ , 当  $j=0$  时,  $X_f^{-1} = X$ 。用 SVM 求取最优分类面  $O^j(x)=0$  的步骤如下。

**Step1:** 求取训练样本。抽取任意点  $x_i \in X_f^{j-1}$  (当  $j=0$  时, 则抽取任意点  $x_i \in X$ ), 并判断与  $x_i$  对应的  $y_i$ , 其判断方法如下:

$$\text{如果 } F(x_i) \geq F_{X_f^j}^*(x_i)$$

$$\text{则, } y_i = +1$$

$$\text{否则, } y_i = -1$$

其中,  $F_{X_f^j}^*(x_i)$  是下述优化问题的极值:

$$\begin{aligned} \min_{u \in U} F_{X_f^j}(x) &= q(x, u) + F(f(x, u)) \\ \text{s.t.} \quad & f(x, u) \in X_f^{j-1} \end{aligned}$$

**Step2:** 将训练样本输入支持向量机。将由 Step1 中求取的每一对  $x_i, y_i$  输入支持向量机。

**Step3:** 求取最优分类面  $O^j(x) = 0$ ，确定  $X_f^j$ 。支持向量机从所有的训练样本中抽取出  $P_j$  个支持向量，并将  $P_j$  个支持向量  $x_i$  和与之对应的权重系数  $w_i$  及分类阈值  $b$  一并输出。保存这些数据，由式 (2-17) 确定最优分类面  $O^j(x) = 0$ 。

$$O^j(x) = \sum_{i=1}^{P_j} w_i \cdot \ker(x_i, x) + b \quad (2-17)$$

其中， $P_j$  为支持向量个数， $w_i$  为支持向量  $x_i$  所对应的权重系数， $b$  为分类阈值， $\ker(x_i, x)$  为高斯核函数，其形式如式 (2-13) 所示。得到  $O^j(x)$  后， $X_f^j$  即可描述为

$$X_f^j \triangleq \{x \in X_f^{j-1} \mid O^j(x) \geq 0\}$$

**第三步:** 判断是否满足循环终止条件。当  $j = N_{\text{stop}}$  时，若满足

$$\sum_{i=1}^{P_{N_{\text{stop}}}} \|O^{N_{\text{stop}}}(x_i) - O^{N_{\text{stop}}-1}(x_i)\| \leq \varepsilon P_{N_{\text{stop}}}$$

则跳出循环。

**第四步:** 获取终端状态集的最终估计。

$$X_{f,\max} \triangleq X_f^{N_{\text{stop}}} = \{x \in X \mid O^{N_{\text{stop}}}(x) \geq 0\}$$

以上就是求取终端状态集的具体步骤。当终端状态集确定后，式 (2-2) 表示的优化问题即可描述为



$$\begin{aligned}
\min_{u(i, x_k) \in U} J(u, x_k) &= \sum_{i=0}^{N-1} q(x(i, x_k), u(i, x_k)) + F(x(N, x_k)) \\
\text{s.t. } x(i+1, x_k) &= f(x(i, x_k), u(i, x_k)) \\
x(i+1, x_k) &\in X, u(i, x_k) \in U, O^{N_{\text{stop}}}(x(N, x_k)) \geq 0
\end{aligned} \tag{2-18}$$

其中,  $O^{N_{\text{stop}}}(x)$  是  $j = N_{\text{stop}}$  时的最优分类函数, 其形式为

$$O^{N_{\text{stop}}}(x) = \sum_{i=1}^{P_{N_{\text{stop}}}} w_i \cdot \ker(x_i, x) + b \tag{2-19}$$

**注 2.5**  $O^{N_{\text{stop}}}(x)$  的获取是一个离线计算过程, 并不参与到预测控制的在线优化中。因此,  $O^{N_{\text{stop}}}(x)$  的获取对时间并没有特别高的要求, 这也是可以采用支持向量机对其进行迭代逼近的原因之一。但是, 可以看出, 由最优分类函数  $O^{N_{\text{stop}}}(x)$  确定的终端状态集并不一定是凸的, 这会导致在预测控制的在线优化计算中, 计算量偏大, 计算时间偏长。如何在此终端状态集内求取一个最大凸集作为新的终端状态集将是我们下一步的研究方向。

**注 2.6** 如式(2-19)所示, 采用此方法求取的终端状态集只与其最优分类函数相关, 而最优分类函数的复杂度与其用到的支持向量个数相关。支持向量个数与迭代次数之间并不存在必然联系, 因此, 最终得到的终端状态集不会随着迭代次数的增加而愈发复杂。

### 2.3.6 仿真算例

将文献[8,37]中的模型离散化后用作本节方法的仿真模型。

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T \\ T & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} T\mu \\ T\mu \end{bmatrix} u_k + \begin{bmatrix} T(1-\mu) & 0 \\ 0 & -4T(1-\mu) \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} u_k$$

其中,  $\mu = 0.5$ ; 状态约束与输入约束分别为  $X = \{x \mid \|x\|_1 \leq 4\}$ ,  $U = \{u \mid |u| \leq 2\}$ ; 采样时间  $T = 0.1\text{s}$ ; 预测步长  $N = 3$ 。在文献[8,37]中, 代价函数取二次型性能指标,  $q(x, u) = x^T Q_0 x + u^T R_0 u$ ,  $Q_0 = 0.5I$ ,  $R_0 = 1$ , 离散形式的代价函数需

将其乘以采样时间，即  $Q = TQ_0$ ,  $R = TR_0$ ；终端代价函数  $F(x) = x^T P x$ ， $P = [345 \ 300; 300 \ 345]$ 。

对于支持向量机来说，训练样本数的选取很重要，太小则精度不高，太大则计算量大。经过多次仿真试验，将训练样本数选定为 3000，计算量较小且精度与更大样本数的精度相差无几。首先选取 3000 个任意点，再对每一个点  $x_i$  判别其相应的  $y_i$ ，最后将  $\{x_i\}, \{y_i\}, i = 1, \dots, 3000$  两组数据输入支持向量机，得到最优分类面。

在式 (2-13) 中，高斯核函数的参数选取为  $\sigma = 1$ ，判别不等式 (2-16) 中的阈值设置为  $\varepsilon = 1$ 。仿真结果显示，当迭代至  $j = 15$  时，已经满足

$$\sum_{i=1}^{P_{15}} \|O^{15}(x_i) - O^{14}(x_i)\| \leq 1 \times P_{15}$$

此时，即可认为  $X_f^{15}$  与  $X_f^{14}$  已经很接近，并将  $X_f^{15}$  作为  $X_{f,\max}$  的最终估计。整个逼近过程如图 2-2 所示。

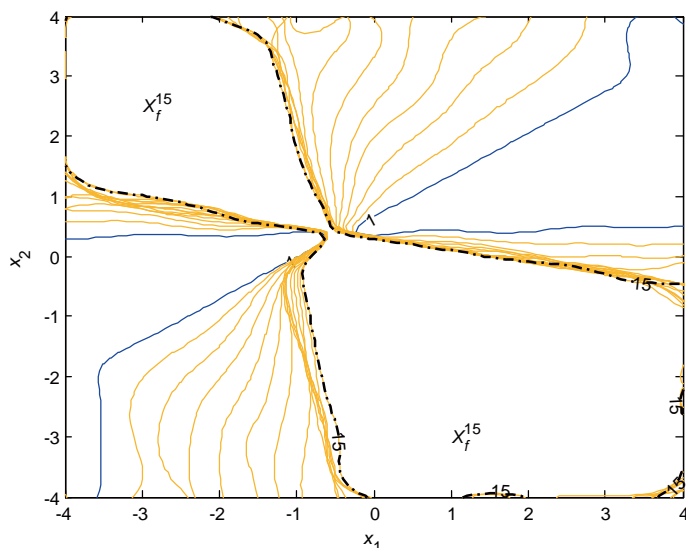


图 2-2 外包集序列的收敛示意图

在图 2-2 中, 标号为“1”的实线(图中左下角由横坐标约为-3.6 的位置出发、到纵坐标约为 0.3 的位置结束的那条曲线, 以及图中右上角由横坐标约为 3.4 的位置出发、到纵坐标约为 0.5 的位置结束的那条曲线, 其在右上角处还有一小部分)表示从状态空间中分离  $X_f^0$  的最优分类面; 标号为“15”的加粗点画线(图中左上角与右下角均标有  $X_f^{15}$  的那片空白区域的边界围线)表示分离  $X_f^{15}$  的最优分类面; 实线“1”与实线“15”之间的实线表示分离  $X_f^j, j=2, \dots, 14$  的最优分类面。确定  $O^{15}(x)$  时用到的支持向量个数为  $P_{15} = 33$ , 保存这 33 个支持向量及其对应的权重系数  $w_i$  及分类阈值  $b$  (因篇幅所限, 这里就不一一给出), 代入式 (2-19), 即可得到  $X_f^{15}$  的近似描述:

$$X_f^{15} \triangleq \left\{ x \in X \mid \sum_{i=1}^{P_{15}} w_i \cdot \ker(x_i, x) + b \geq 0 \right\}$$

$X_f^{15}$  便可作为优化问题式 (2-18) 中的终端状态约束集。取任意点  $x \in \Gamma_3$ 。本例中,  $\Gamma_3$  与  $X_f^{15}$  非常接近, 因此, 我们任意选取  $x \in X_f^{15}$ , 观察其收敛性, 如图 2-3 所示。

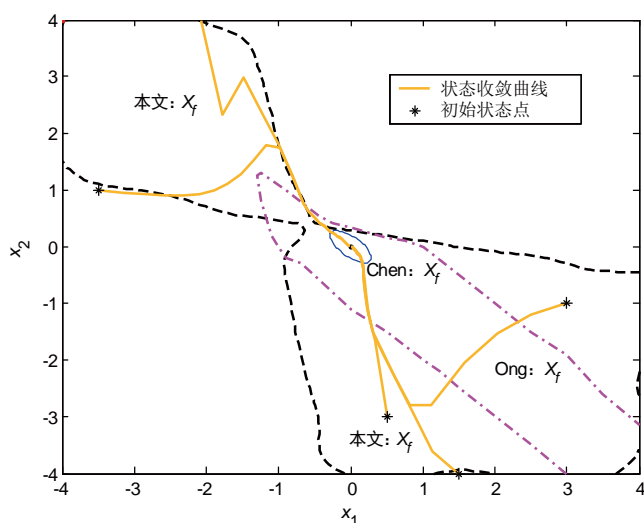


图 2-3 状态收敛曲线

在图 2-3 中, 位于图中心处的细实线椭圆区域表示文献[8]中 Chen 给出的终端状态约束集, 点画线区域表示文献[37]中 Ong 给出的终端状态约束集, 本节方法得到的终端状态约束集用虚线区域表示。实线表示状态点的收敛曲线, 可以看出, 状态点都可被引导至平衡点, 而且采用此方法得到的终端状态约束集比文献[8]和文献[37]得到的结果都大。

## 2.4 用子集序列逼近最大终端状态集

用外包集序列逼近最大终端状态集的逼近策略是由大到小, 即由整个状态空间向最大终端状态集逼近。这种方法存在一个不足之处: 得到的只是最大终端状态集的一个外包集, 与此外包集对应的吸引域可能包含一些非稳定点。对此, 可采用一种思想与此类似但由小到大的逼近策略, 即由最大终端状态集的一个已知子集向其逼近的迭代策略, 得到越来越大的子集。

与 2.3 节介绍的方法类似, 采用由小到大的迭代逼近策略对  $X_f$  进行估计。首先给出一个最大终端状态集的初始子集  $X_f^0$ , 再借由  $X_f^0$  得到一个更大的子集  $X_f^1$ , 如此反复, 依次得到  $X_f^2, X_f^3, \dots$ , 满足  $X_f^0 \subseteq X_f^1 \subseteq \dots \subseteq X_f^j \subseteq X_{f,\max}$ 。随着  $j$  的增大,  $X_f^j$  逐渐收敛到某一集合  $X_f^{+\infty}$ , 此  $X_f^{+\infty}$  即  $X_{f,\max}$  的最终估计。

### 2.4.1 子集序列的构造方法

用子集序列逼近最大终端状态集的示意图如图 2-4 所示。

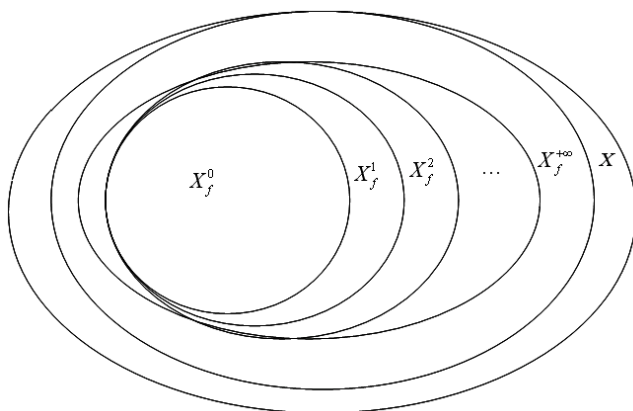


图 2-4 用子集序列逼近最大终端状态集的示意图

与用外包集序列逼近最大终端状态集的逼近策略不同，用子集序列逼近最大终端状态集得到的状态区域仍然是最大终端状态集的子集，对系统稳定性没有影响。但是，这种方法需要一个前提，即已经知道最大终端状态集的某一子集。

**假设 2.1**  $X_{f,\max}$  的一个包含原点的不变子集是已知的，将此子集记为  $X_f^0$ 。

借由  $X_f^0$ ，我们可以构造出  $X_{f,\max}$  的另一个子集  $X_f^1$ 。

$$X_f^1 := \{x \in X \mid F(x) \geq F_{X_f^0}^*(x)\} \quad (2-20)$$

其中， $F_{X_f^0}^*(x)$  是下述优化问题的极小值：

$$\begin{aligned} \min_{u \in U} F_{X_f^0}(x) &= q(x, u) + F(f(x, u)) \\ \text{s.t.} \quad & f(x, u) \in X_f^0 \end{aligned} \quad (2-21)$$

很明显， $X_f^0 \subseteq X_f^1$ 。类似地，另一个子集  $X_f^2$  可以通过相同的方法构造出来。

$$X_f^2 := \{x \in X \mid F(x) \geq F_{X_f^1}^*(x)\} \quad (2-22)$$

其中,  $F_{X_f^j}^*(x)$  是下述优化问题的极小值:

$$\begin{aligned} \min_{u \in U} F_{X_f^j}(x) &= q(x, u) + F(f(x, u)) \\ \text{s.t.} \quad & f(x, u) \in X_f^1 \end{aligned} \quad (2-23)$$

如此反复,  $X_f^j, j=3, 4, \dots, \infty$  都可以被构造出来。

$$X_f^j := \{x \in X \mid F(x) \geq F_{X_f^{j-1}}^*(x)\} \quad (2-24)$$

其中,  $F_{X_f^{j-1}}^*(x)$  是下述优化问题的极小值:

$$\begin{aligned} \min_{u \in U} F_{X_f^{j-1}}(x) &= q(x, u) + F(f(x, u)) \\ \text{s.t.} \quad & f(x, u) \in X_f^{j-1} \end{aligned} \quad (2-25)$$

## 2.4.2 子集序列的收敛性分析

通过 2.4.1 节的方法构造得到的  $\{X_f^j\}$  显然满足  $X_f^{j-1} \subseteq X_f^j$ , 随着迭代次数  $j$  的增加,  $\{X_f^j\}$  将会收敛至某一集合  $X_f^{+\infty}$  (如果  $\{X_f^j\}$  不收敛,  $X_f^{+\infty}$  就与  $X$  完全重合)。以下结论将告诉我们,  $X_f^{+\infty}$  就是我们所要求取的最大终端状态约束集。

**定理 2.3** 当迭代次数  $j$  趋近于无穷大时, 通过上述方法构造得到的集合序列  $\{X_f^j\}$  将收敛至  $X_{f, \max}$ 。

证明: 此处的证明与定理 2.1 的证明类似, 都用反证法。

(I) 假设存在  $X_{\text{spo}} \subset X_{f, \max}$ , 使得  $j \rightarrow +\infty$  时,  $X_f^j \rightarrow X_{\text{spo}}$ 。由假设 2.1 可知,  $0 \in X_f^0$ 。因为  $X_f^0 \subseteq X_{\text{spo}}$ , 所以明显有

$$0 \in X_{\text{spo}}, \quad 0 \notin X_{f, \max} \setminus X_{\text{spo}}$$

因此, 对任意的  $x_0 \in X_{f, \max} \setminus X_{\text{spo}}$ , 都有  $F(x) \geq \xi$ , 其中,  $\xi$  是  $\{F(x) \mid x \in X_{f, \max} \setminus X_{\text{spo}}\}$  的下确界。

另外, 对任意的  $x_0 \in X_{f,\max} \setminus X_{\text{spo}}$ , 不存在这样一个  $u \in U$ , 使得

$$\begin{aligned} F(x_0) &\geq q(x_0, u) + F(f(x_0, u)) \\ f(x_0, u) &\in X_{\text{spo}} \end{aligned}$$

成立。但是, 由条件 (C2) 和 (C3) 可知,  $\exists u(x_0) \in U$ , 使得如下两式被满足

$$\begin{aligned} F(x_0) &\geq q(x_0, u(x_0)) + F(x_1) \\ f(x_1) &\in X_{f,\max} \setminus X_{\text{spo}} \end{aligned}$$

其中

$$x_1 = f(x_0, u(x_0))$$

类似地, 存在  $u(x_i) \in U, i=1, 2, \dots, \infty$ , 使得如下两式被满足

$$\begin{aligned} F(x_i) &\geq q(x_i, u(x_i)) + F(x_{i+1}) \\ f(x_{i+1}) &\in X_{f,\max} \setminus X_{\text{spo}} \end{aligned}$$

其中

$$x_{i+1} = f(x_i, u(x_i))$$

可以看出,  $F(\cdot)$  是一个李雅普诺夫函数。因此, 对任意的  $\delta > 0$ , 都存在  $N > 0$ , 使得对任意的  $i \geq N$ , 都满足

$$\begin{aligned} F(x_i) &< \delta \\ x_i &\in X_{f,\max} \setminus X_{\text{spo}} \end{aligned}$$

选择一个  $\delta$ , 使其满足  $\delta < \xi$ , 则存在  $N_\xi > 0$ , 使如下两式被满足

$$\begin{aligned} F(x_{N_\xi}) &< \delta < \xi \\ x_{N_\xi} &\in X_{f,\max} \setminus X_{\text{spo}} \end{aligned}$$

显然, 这与  $\xi$  是  $\{F(x) | x \in X_{f,\max} \setminus X_{\text{spo}}\}$  的下确界相矛盾。

(II) 假设存在  $X_{\text{spo}} \supset X_{f,\max}$ , 使得  $j \rightarrow +\infty$  时,  $X_f^j \rightarrow X_{\text{spo}}$ 。对任意的  $x \in X_{\text{spo}}$ , 有

$$F(x) \geq \min_{u \in U} \{q(x, u) + F(f(x, u))\}$$

$$f(x, u) \in X_{\text{spo}}$$

成立。

很明显, 这与  $X_{f,\max}$  是满足条件 (C2) 和 (C3) 的最大终端状态约束集相矛盾。

证毕。

**注 2.7** 与用外包集序列来逼近  $X_{f,\max}$  类似, 在实际计算过程中, 因时间有限, 是无法计算到  $j \rightarrow +\infty$  的。若迭代到  $j = N_{\text{stop}}$  时,  $X_f^{N_{\text{stop}}}$  与  $X_f^{N_{\text{stop}}-1}$  很接近, 即可将  $X_f^{N_{\text{stop}}}$  作为对  $X_{f,\max}$  的最终估计。由此方法得到的  $X_f^{N_{\text{stop}}}$  本质上是  $X_{f,\max}$  的一个子集, 不会影响预测控制的稳定性。因此, 用子集序列来逼近  $X_{f,\max}$  相较于用外包集序列来逼近  $X_{f,\max}$  更加优越一些。

### 2.4.3 用子集序列逼近最大终端状态集的计算步骤

与用外包集序列逼近最大终端状态集类似, 子集序列中的各个子集可用支持向量机从状态空间中一个一个分离估计出来, 具体步骤可整理如下。

**第一步:** 设置 SVM 的训练样本数及循环终止阈值  $\varepsilon$ , 并给定包含原点的初始子集  $X_f^0$ 。

**第二步:** 导入循环  $j=1, 2, 3, \dots$ , 用 SVM 求取最优分类面  $O^j(x)=0$ , 确定  $X_f^j \triangleq \{x \in X_f^{j-1} | O^j(x) \geq 0\}$ 。用 SVM 求取最优分类面  $O^j(x)=0$  的步骤如下。



**Step1:** 求取训练样本。抽取任意点  $x_i \in X$ ，并判断与  $x_i$  对应的  $y_i$ ，其判断方法如下：

$$\text{如果 } F(x_i) \geq F_{X_f^j}^*(x_i)$$

$$\text{则, } y_i = +1$$

$$\text{否则, } y_i = -1$$

其中,  $F_{X_f^j}^*(x_i)$  是下述优化问题的极值：

$$\begin{aligned} \min_{u \in U} F_{X_f^j}(x) &= q(x, u) + F(f(x, u)) \\ \text{s.t. } f(x, u) &\in X_f^{j-1} \end{aligned}$$

**Step2:** 将训练样本输入支持向量机。将由 Step1 中求取的每一对  $x_i, y_i$  输入支持向量机。

**Step3:** 求取最优分类面  $O^j(x) = 0$ ，确定  $X_f^j$ 。支持向量机从所有的训练样本中抽取出  $P_j$  个支持向量，并将  $P_j$  个支持向量  $x_i$  和与之对应的权重系数  $w_i$  及分类阈值  $b$  一并输出。保存这些数据，确定最优分类面  $O^j(x) = 0$ 。

$$O^j(x) = \sum_{i=1}^{P_j} w_i \cdot \ker(x_i, x) + b$$

其中,  $P_j$  为支持向量个数,  $w_i$  为支持向量  $x_i$  所对应的权重系数,  $b$  为分类阈值,  $\ker(x_i, x)$  为高斯核函数, 其形式为

$$\begin{aligned} \phi(x)\phi(x_i) &= \ker(x_i, x) \\ &= \exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) \end{aligned}$$

得到  $O^j(x)$  后,  $X_f^j$  即可描述为

$$X_f^j \triangleq \{x \in X_f^{j-1} \mid O^j(x) \geq 0\}$$

**第三步：**判断是否满足循环终止条件。当  $j = N_{\text{stop}}$  时，若满足

$$\sum_{i=1}^{P_{N_{\text{stop}}}} \|O^{N_{\text{stop}}}(x_i) - O^{N_{\text{stop}}-1}(x_i)\| \leq \varepsilon P_{N_{\text{stop}}}$$

则跳出循环。

**第四步：**获取终端状态集的最终估计。

$$X_{f,\max} \triangleq X_f^{N_{\text{stop}}} = \{x \in X \mid O^{N_{\text{stop}}}(x) \geq 0\}$$

以上就是求取终端状态集的具体步骤。当终端状态集确定后，预测控制的优化问题即可描述为

$$\begin{aligned} \min_{u(i, x_k) \in U} J(u, x_k) &= \sum_{i=0}^{N-1} q(x(i, x_k), u(i, x_k)) + F(x(N, x_k)) \\ \text{s.t. } x(i+1, x_k) &= f(x(i, x_k), u(i, x_k)) \\ x(i+1, x_k) &\in X, u(i, x_k) \in U, O^{N_{\text{stop}}}(x(N, x_k)) \geq 0 \end{aligned}$$

其中， $O^{N_{\text{stop}}}(x)$  是  $j = N_{\text{stop}}$  时的最优分类函数，其形式为

$$O^{N_{\text{stop}}}(x) = \sum_{i=1}^{P_{N_{\text{stop}}}} w_i \cdot \ker(x_i, x) + b$$

## 2.4.4 仿真算例

此处的仿真模型与 2.3.6 节一致，各仿真条件也一致， $X_{f,\max}$  的子集序列的初始子集选用文献[8]的终端状态约束集，即

$$X_f^0 = \left\{ x \in X \mid x^T \begin{bmatrix} 16.5926 & 11.5926 \\ 11.5926 & 16.5926 \end{bmatrix} x \leq 0.7 \right\}$$

采用支持向量机进行子集估计，当迭代至  $j = 22$  时， $\hat{X}_f^{22}$  与  $\hat{X}_f^{21}$  很接近，因此可将  $\hat{X}_f^{22}$  作为  $X_{f,\max}$  的最终估计。整个逼近过程如图 2-5 所示。

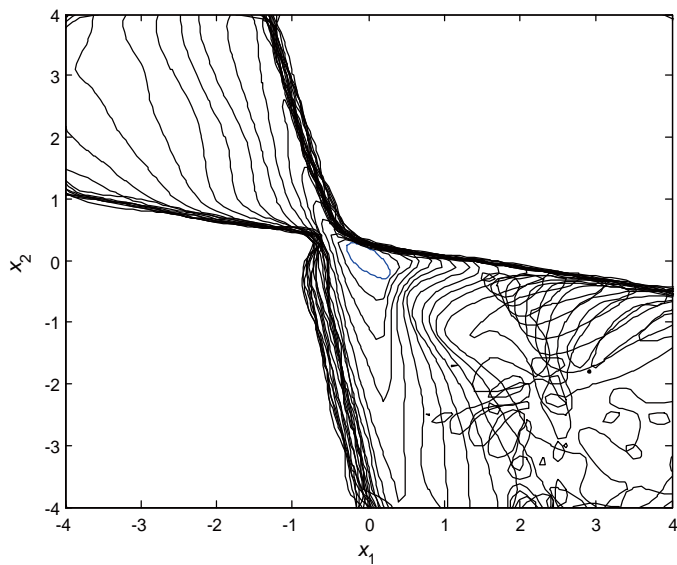


图 2-5 子集序列的收敛示意图

同样，从吸引域中选取任意点，观察其收敛性，如图 2-6 所示。

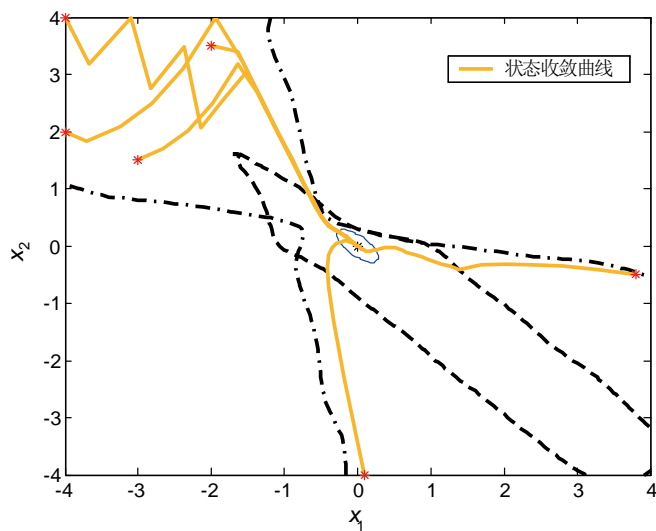


图 2-6 状态收敛曲线

在图 2-5 中,位于图中心位置的椭球区域表示文献[8]的终端状态约束集,它是这次仿真中选择的初始子集;另外 22 条实线描述了由文献[8]的终端状态约束集向最大终端状态约束集逐渐逼近的过程。

在图 2-6 中,位于图中心位置的椭球区域表示文献[8]的终端状态约束集,虚线围起来的区域表示文献[37]的结果,点画线围起来的区域表示采用本节方法的终端状态约束集,另外几条实线为状态的收敛曲线。

## 2.5 本章小结

本章给出了两种求取最大终端状态约束集的方法,采用这两种方法得到的终端状态约束集相较于现有文献得到的结果要大一些,这对增大预测控制的吸引域具有一定的意义。其中,第一种方法采用以外包集序列逼近最大终端状态约束集的策略,第二种方法采用以子集序列逼近最大终端状态约束集的策略。从理论上证明了两种方法随着迭代次数的增加,状态集序列都逐渐收敛于最大终端状态集。两种方法都以支持向量机为分类工具从状态空间中依次分离出状态集序列,由图 2-2 与图 2-5 可以看出,这两种方法得到的终端状态集几乎相同。但是,从原理上讲,第一种方法对预测控制的稳定性有所影响,而第二种方法却没有。

## 第 3 章

# 求取预测控制的终端代价函数

### 3.1 引言

由条件 (C2) 和 (C3) 可以看出, 预测控制的终端状态约束集与终端代价函数之间有着密切的关系。而第 2 章求取终端状态约束集的方法都是基于终端代价函数已知这个前提的, 因此, 有必要研究如何求取预测控制的终端代价函数。第 1 章已经提到, 除了与终端状态约束集有密切的关系, 终端代价函数还与预测控制所消耗的全局性能指标有关。但是, 迄今为止, 关于终端代价函数的这些性质的研究却很少见。文献[38]针对无终端状态约束预测控制证明了按比例增大终端代价函数也会增大预测控制的吸引域; 文献[39]重点分析了无终端状态约束的预测控制的全局性能指标, 给出了全局性能指标的上下界; 文献[40]设计了一种预测控制算法, 其标称消耗的全局性能指标与最优性能指标一致, 但其终端代价函数中  $F(x) = C_f$  本质上是一种终端等式约束。

如前所述, 衡量一种预测控制算法的好坏主要看两方面因素: 吸引域的大小 (在预测步长一定的前提下, 取决于终端状态约束集的大小); 全局性

能指标消耗的多少（采用滚动控制律将初始状态控制至平衡点所消耗的性能指标总和）。因此，终端代价函数的选择对决定预测控制的控制性能的好坏有着很关键的作用，而衡量预测控制的控制性能的好坏则主要是以最优控制为参考的。

本章首先给出预测控制与最优控制的关系；接着分析预测控制的终端代价函数分别与终端状态约束集和全局性能指标之间的定性关系，以此定性关系为基础，给出一种简单的求取预测控制终端代价函数的方法；最后提出一种具有自调节终端代价函数的模型预测控制方案，使终端代价函数能随着控制过程的推进而进行适当的调整，可用较少的全局性能指标消耗保证闭环系统的性能。

## 3.2 预测控制与最优控制的关系

考虑如下系统 [此系统即第 2 章中的系统式 (2-1)]:

$$x_{k+1} = f(x_k, u_k) \quad (3-1)$$

最优控制在初始点  $x_0$  的优化问题  $P_\infty(x_0)$  可被描述为

$$\begin{aligned} \min_{u(i, x_0) \in U} J_\infty(u, x_0) &= \sum_{i=0}^{\infty} q(x(i, x_0), u(i, x_0)) \\ \text{s.t. } x(i+1, x_0) &= f(x(i, x_0), u(i, x_0)) \\ x(i+1, x_0) &\in X, u(i, x_0) \in U \end{aligned} \quad (3-2)$$

其中， $x(0, x_0) = x_0$ ， $x_0$  表示优化起始点； $X$  和  $U$  分别为状态约束与输入约束； $q(x, u)$  为形式为  $q(x, u) = x^T Q x + u^T R u$  的阶段指标函数，其中  $Q$  和  $R$  分别为对应维数的正定矩阵。

如果优化问题  $P_\infty(x_0)$  有解，则定义  $J_\infty^*(x_0)$  为  $J_\infty(u, x_0)$  的极小值，最优

输入序列为

$$\mathbf{u}_\infty^*(x_0) = \{u_\infty^*(0, x_0), u_\infty^*(1, x_0), u_\infty^*(2, x_0), \dots, u_\infty^*(+\infty, x_0)\} \quad (3-3)$$

最优预测状态轨迹为

$$\mathbf{x}_\infty^*(x_0) = \{x_\infty^*(1, x_0), x_\infty^*(2, x_0), x_\infty^*(3, x_0), \dots, x_\infty^*(+\infty, x_0)\} \quad (3-4)$$

众所周知, 如果优化问题  $P_\infty(x_0)$  有解, 则采用最优控制消耗的全局性能指标就是  $J_\infty^*(x_0)$ , 而最优控制的吸引域  $\Gamma_\infty$  则可定义为所有满足优化问题式 (3-2) 有解的状态点的集合, 即

$$\Gamma_\infty := \{x \in X \mid P_\infty(x) \text{ 有解}\} \quad (3-5)$$

但是, 求解优化问题  $P_\infty(x_0)$  是很难的, 尤其是当  $f(\cdot, \cdot)$  为非线性函数的时候。而预测控制则可以避免这个问题, 从这个层面上讲, 预测控制可以理解为用基于有限时域优化 (或半无限时域优化<sup>[8,9]</sup>) 的次优控制算法来逼近基于无限时域优化的最优控制算法的一种控制算法。

以带终端状态约束集和终端代价函数的预测控制为例, 其优化问题  $P_N(x_0)$  可描述为

$$\begin{aligned} \min_{\mathbf{u}(i, x_0) \in U} J(\mathbf{u}, x_0) &= \sum_{i=0}^{N-1} q(x(i, x_0), u(i, x_0)) + F(x(N, x_0)) \\ \text{s.t. } x(i+1, x_0) &= f(x(i, x_0), u(i, x_0)) \\ x(i+1, x_0) &\in X, u(i, x_0) \in U, i = 0, \dots, N-1 \\ x(N, x_0) &\in X_f \end{aligned} \quad (3-6)$$

其中,  $X_f$  为终端状态约束集;  $F(\cdot)$  为终端代价函数。

定义  $J^*(x_0)$  为  $J(\mathbf{u}, x_0)$  的极小值,  $\mathbf{u}^*(x_0) = \{u^*(0, x_0), \dots, u^*(N-1, x_0)\}$  为其最优输入序列,  $\mathbf{x}^*(x_0) = \{x^*(1, x_0), \dots, x^*(N, x_0)\}$  为相应的最优预测状态轨迹。在实际控制中, 采取的是滚动控制策略, 即真正输入实际系统的是滚

动控制律

$$\mathbf{u}_{\text{RH}} = \{u^*(0, x_0), u^*(0, x_1), u^*(0, x_2), \dots\} \quad (3-7)$$

采用预测控制消耗的全局性能指标即可描述为

$$J_{\text{RH}}(x_0) = \sum_{i=0}^{\infty} q(x_i, u^*(0, x_i)) \quad (3-8)$$

考虑到阶段代价函数的形式，预测控制消耗的全局性能指标可进一步表示为

$$J_{\text{RH}}(x_0) = \sum_{i=0}^{+\infty} x_i^T \mathbf{Q} x_i + u^*(0, x_i)^T \mathbf{R} u^*(0, x_i) \quad (3-9)$$

其中

$$x_{i+1} = f(x_i, u^*(0, x_i)), \quad i = 0, 1, \dots, +\infty \quad (3-10)$$

第2章中已经定义了预测控制的吸引域，即

$$\Gamma_N := \{x \in X \mid x^*(N, x) \in X_f\} \quad (3-11)$$

前面已经提到，为保证预测控制系统的闭环稳定性，终端状态集  $X_f$  需要满足 (C1)、(C2) 和 (C3) 三个条件。

理想的终端状态集为

$$X_f = \Gamma_{\infty} \quad (3-12)$$

理想的终端代价函数为

$$F(\cdot) = J_{\infty}^*(\cdot) \quad (3-13)$$

如果式 (3-12) 与式 (3-13) 都能满足，则预测控制的优化问题可描述为



$$\begin{aligned}
\min_{u(i, x_0) \in U} J(u, x_0) &= \sum_{i=0}^{N-1} q(x(i, x_0), u(i, x_0)) + J^*(x(N, x_0)) \\
\text{s.t. } x(i+1, x_0) &= f(x(i, x_0), u(i, x_0)) \\
x(i+1, x_0) &\in X, u(i, x_0) \in U, i=0, \dots, N-1 \\
x(N, x_0) &\in \Gamma_\infty
\end{aligned} \tag{3-14}$$

此时, 预测控制的吸引域为

$$\Gamma_N = \Gamma_\infty \tag{3-15}$$

而其消耗的全局性能指标则为

$$J_{\text{RH}}(\cdot) = J_\infty^*(\cdot) \tag{3-16}$$

**注 3.1** 当式 (3-12) 与式 (3-13) 都能满足时, 预测控制的吸引域与其预测步长之间没有必然的联系。

但是, 正如前面内容所示, 求解优化问题  $P_\infty(x_0)$  是很难的, 尤其是当  $f(\cdot, \cdot)$  为非线性函数的时候,  $P_\infty(x_0)$  的求解几乎是不可能的。因此,  $J_\infty^*(\cdot)$  是得不到的,  $\Gamma_\infty$  也是得不到的。也正是由于这一点, 使得我们所设计的预测控制算法的吸引域总是小于最优控制的吸引域, 即

$$\Gamma_N \leq \Gamma_\infty \tag{3-17}$$

而预测控制所消耗的全局性能指标也总是大于最优控制的最优指标, 即

$$J_{\text{RH}}(\cdot) \geq J_\infty^*(\cdot) \tag{3-18}$$

因此, 在设计预测控制算法时, 总是希望预测控制的吸引域  $\Gamma_N$  尽量大 (在预测步长一定的前提下, 终端状态集  $X_f$  尽量大), 消耗的全局性能指标  $J_{\text{RH}}(\cdot)$  尽量少。但实际上, 兼得这两者却是很难的。3.3 和 3.4 节将着重讨论这一问题, 分析预测控制的终端代价函数与终端状态约束集以及终端代价函数与全局性能指标之间的定性关系。

### 3.3 终端代价函数与终端状态集之间的关系

对于系统式 (3-1) 和预测控制式 (3-6), 考虑按比例增大预测控制的终端代价函数, 其优化问题定义为  $P_{N,\lambda}(x_0)$ , 具体描述如下:

$$\begin{aligned} \min_{u(i,x_0) \in U} J(u, x_0) &= \sum_{i=0}^{N-1} q(x(i, x_0), u(i, x_0)) + \lambda F(x(N, x_0)) \\ \text{s.t. } x(i+1, x_0) &= f(x(i, x_0), u(i, x_0)) \\ x(i+1, x_0) &\in X, u(i, x_0) \in U, i = 0, \dots, N-1 \\ x(N, x_0) &\in X_{\lambda f} \end{aligned} \quad (3-19)$$

其中,  $\lambda F(\cdot)$  为终端代价函数,  $\lambda \geq 1$  为比例系数,  $X_{\lambda f}$  为与终端代价函数  $\lambda F(\cdot)$  对应的终端状态约束集。

定义  $X_{\lambda f, \max}$  为以  $P_{N,\lambda}(x_0)$  描述的预测控制的最大终端状态约束集。文献[38]针对在线优化中不带显式终端状态约束的预测控制 (第 1 章中讲到的第三种预测控制) 证明了按比例增大终端代价函数, 预测控制的终端状态集 (对第三种预测控制来说, 即其在线优化末端状态会自动进入的原点邻域  $\Omega$ ) 会增大。本节将证明对于带终端代价函数与终端状态约束的预测控制来说, 这一性质仍然成立。

**定理 3.1** 考虑终端代价函数  $F(\cdot)$  和与之对应的最大终端状态约束集  $X_{f, \max}$ , 按比例增大终端代价函数至  $\lambda F(\cdot)$ , 当  $\lambda \geq 1$  时, 其对应的最大终端状态约束集  $X_{\lambda f, \max}$  也会增大, 即  $X_{f, \max} \subseteq X_{\lambda f, \max}$ 。

证明:

对任意的  $x_0 \in X_{f, \max}$  来说, 满足条件 (C1) (C2) 和 (C3) 是指存在一个控制序列  $u = \{u_0, u_1, u_2, \dots\}, u_i \in U$ , 使得下面的不等式成立:

$$F(x_i) \geq q(x_i, u_i) + F(x_{i+1}), i = 0, 1, 2, \dots$$

其中

$$x_{i+1} = f(x_i, u_i)$$

很明显, 当终端代价函数为  $\lambda F(\cdot)$  时, 采用同样的控制序列, 下面的不等式也成立:

$$\lambda F(x_i) \geq q(x_i, u_i) + \lambda F(x_{i+1})$$

可知

$$x_0 \in X_{\lambda f, \max}$$

因此

$$X_{f, \max} \subseteq X_{\lambda f, \max}$$

证毕。

**注 3.1** 由定理 3.1 可以看出, 按比例增大终端代价函数, 预测控制的最大终端状态约束集  $X_{\lambda f, \max}$  也会增大。因此, 当预测步长一定时, 预测控制的吸引域也会增大。

### 3.4 终端代价函数与全局性能指标之间的关系

由 3.3 节可以看出, 按比例增大终端代价函数, 预测控制的最大终端状态约束集也会增大, 这是否意味着我们可以随意增大终端代价函数呢? 答案是否定的, 本节将给出其具体原因。

定义  $J_{N, \lambda}^*(x_0)$  为优化问题  $P_{N, \lambda}(x_0)$  的极小值, 最优控制序列为

$$\mathbf{u}_{N,\lambda}^*(x_0) = \{u_{N,\lambda}^*(0, x_0), \dots, u_{N,\lambda}^*(N, x_0)\} \quad (3-20)$$

最优预测状态轨迹为

$$\mathbf{x}_{N,\lambda}^*(x_0) = \{x_{N,\lambda}^*(0, x_0), \dots, x_{N,\lambda}^*(N, x_0)\} \quad (3-21)$$

预测控制的滚动控制轨迹为

$$\mathbf{u}_{\text{RH},\lambda}(x_0) = \{u_N^*(0, x_0), u_N^*(0, x_{1,\lambda}), \dots\} \quad (3-22)$$

滚动状态轨迹为

$$\mathbf{x}_{\text{RH},\lambda}(x_0) = \{x_0, x_{1,\lambda}, x_{2,\lambda}, \dots\} \quad (3-23)$$

其中

$$x_{i+1,\lambda} = f(x_{i,\lambda}, u_N^*(0, x_{i,\lambda})), \quad i = 1, 2, \dots, +\infty$$

$$x_{0,\lambda} = x_0$$

预测控制消耗的全局性能指标为

$$J_{\text{RH},\lambda}(x_0) = \sum_{i=0}^{+\infty} q(x_{i,\lambda}, u_N^*(0, x_{i,\lambda})) \quad (3-24)$$

考虑到阶段代价函数的形式，全局性能指标可进一步表示为

$$J_{\text{RH}}(x_0) = \sum_{i=0}^{+\infty} x_{i,\lambda}^T \mathbf{Q} x_{i,\lambda} + u^*(0, x_{i,\lambda})^T \mathbf{R} u^*(0, x_{i,\lambda}) \quad (3-25)$$

**假设 3.1** 对任意的  $x_0 \in \Gamma_N$ ，其中， $\Gamma_N$  为以优化问题  $P_N(x_0)$  [如式(3-6)所示] 描述的预测控制的吸引域，求解优化问题  $P_{N,\lambda}(x_0)$  得到的终端状态落入  $X_{f,\max}$  内，即  $x_{N,\lambda}^*(N, x_0) \in X_{f,\max}$ 。

**注 3.2** 假设 3.1 成立则意味着，对任意的  $x_0 \in \Gamma_N$ ，将优化问题  $P_N(x_0)$  [如式(3-19)所示] 中的终端状态约束集  $X_{\lambda f, \max}$  替换成  $X_{f, \max}$  后，其最优解不变。对于  $\Gamma_N$  内的大多数点来说，假设 3.1 是成立的，只有极少数点可能会

不满足，为方便起见，我们忽略了这些少数点的影响。

当假设 3.1 成立时，显然有  $J_{N,\lambda}^*(x_0) \geq J_N^*(x_0)$ ，而且下述引理也很容易得到。

**引理 3.1** 若假设 3.1 成立，则对任意的  $x_0 \in \Gamma_N$ ，下述结论成立：

$$(J_{N-1,\lambda}^* - J_{N-1}^*)(x_N^*(1, x_0)) \geq (J_{N-1,\lambda}^* - J_{N-1}^*)(x_{N,\lambda}^*(1, x_0)) \quad (3-26)$$

证明：

由最优性可知， $J_N^*(x_0)$  可被描述为

$$J_N^*(x_0) = q(x_0, u_N^*(0, x_0)) + J_{N-1}^*(x_N^*(1, x_0)) \quad (3-27)$$

由假设 3.1 和最优性可知

$$J_N^*(x_0) \leq q(x_0, u_{N,\lambda}^*(0, x_0)) + J_{N-1}^*(x_{N,\lambda}^*(1, x_0)) \quad (3-28)$$

类似地， $J_{N,\lambda}^*(x_0)$  可被描述为

$$J_{N,\lambda}^*(x_0) = q(x_0, u_{N,\lambda}^*(0, x_0)) + J_{N-1,\lambda}^*(x_{N,\lambda}^*(1, x_0)) \quad (3-29)$$

同理，由最优性可知

$$J_{N,\lambda}^*(x_0) \leq q(x_0, u_N^*(0, x_0)) + J_{N-1,\lambda}^*(x_N^*(1, x_0)) \quad (3-30)$$

显然，将不等式 (3-30) 的右边减去不等式 (3-28) 的左边得到的结果要大于不等式 (3-28) 的右边减去不等式 (3-30) 的左边，即

$$\begin{aligned} & (q(x_0, u_N^*(0, x_0)) + J_{N-1,\lambda}^*(x_N^*(1, x_0))) - J_N^*(x_0) \geq \\ & J_{N,\lambda}^*(x_0) - (q(x_0, u_{N,\lambda}^*(0, x_0)) + J_{N-1}^*(x_{N,\lambda}^*(1, x_0))) \end{aligned} \quad (3-31)$$

因此有

$$(J_{N-1,\lambda}^* - J_{N-1}^*)(x_N^*(1, x_0)) \geq (J_{N-1,\lambda}^* - J_{N-1}^*)(x_{N,\lambda}^*(1, x_0))$$

证毕。

定义  $F_{\mathcal{K}}$  为这样一类  $\mathcal{K}$  类函数：对任意的  $F_1, F_2 \in F_{\mathcal{K}}$  和  $x_1, x_2 \in X$ ，如果  $F_1(x_1) \geq F_1(x_2)$  成立，则  $F_2(x_1) \geq F_2(x_2)$  也成立。

**假设 3.2** 本章用到的所有代价函数，如  $J_{N-1}^*$ 、 $J_N^*$ 、 $J_{N-1,\lambda}^*$ 、 $J_{N,\lambda}^*$ ，以及其正定的和差运算，如  $J_{N-1,\lambda}^* - J_{N-1}^*$ 、 $J_{N-1,\lambda}^* + J_N^*$ ，都属于  $F_{\mathcal{K}}$ 。

当假设 3.2 成立时，由引理 3.1 可知，对任意的  $x_0 \in \Gamma_N$ ，都有

$$F_{\mathcal{K}}(x_N^*(1, x_0)) \geq F_{\mathcal{K}}(x_{N,\lambda}^*(1, x_0)) \quad (3-32)$$

基于不等式 (3-32)，我们可以得到另一条关键结论。

**引理 3.2** 当假设 3.1 和假设 3.2 成立时，对任意的  $x_0 \in \Gamma_N$ ，任意的正定代价函数  $F_{\text{tra}} \in F_{\mathcal{K}}$ ，若满足  $F_{\text{tra}} \leq J_{N-1}^*$ ，则有

$$q(x_0, u_N^*(0, x_0)) + F_{\text{tra}}(x_1) \leq q(x_0, u_{N,\lambda}^*(0, x_0)) + F_{\text{tra}}(x_{1,\lambda}) \quad (3-33)$$

证明：

首先明确， $x_1$  意指  $x_N^*(1, x_0)$ ， $x_{1,\lambda}$  意指  $x_{N,\lambda}^*(1, x_0)$ 。若假设 3.2 成立，显然有  $(J_{N-1}^* - F_{\text{tra}}) \in F_{\mathcal{K}}$ ，因此，我们可以得到

$$\begin{aligned} & q(x_0, u_N^*(0, x_0)) + F_{\text{tra}}(x_1) \\ &= q(x_0, u_N^*(0, x_0)) + J_{N-1}^*(x_1) - (J_{N-1}^* - F_{\text{tra}})(x_1) \\ &\leq q(x_0, u_{N,\lambda}^*(0, x_0)) + J_{N-1}^*(x_{1,\lambda}) - (J_{N-1}^* - F_{\text{tra}})(x_{1,\lambda}) \\ &= q(x_0, u_{N,\lambda}^*(0, x_0)) + F_{\text{tra}}(x_{1,\lambda}) \end{aligned} \quad (3-34)$$

在此处的证明中，我们用到了两个结论，即

$$\begin{aligned} J_N^*(x_0) &\leq J_{N,\lambda}^*(x_0) \\ (J_{N-1}^* - F_{\text{tra}})(x_1) &\geq (J_{N-1}^* - F_{\text{tra}})(x_{1,\lambda}) \end{aligned}$$

证毕。

至此,不能随意增大终端代价函数的原因就可以给出,如定理 3.2 所示。

**定理 3.2** 当假设 3.1 和假设 3.2 成立时,对任意的  $x_0 \in \Gamma_N$ , 都有

$$J_{RH}(x_0) \leq J_{RH,\lambda}(x_0) \quad (3-35)$$

证明:

很明显,因为  $J_N^* \leq J_{N-1}^*$ , 所以  $J_N^*$  满足引理 3.2 中的条件。选择引理 3.2 中的  $F_{tra} = J_N^*$ , 有

$$q(x_0, u_N^*(0, x_0)) + J_N^*(x_1) \leq q(x_0, u_{N,\lambda}^*(0, x_0)) + J_N^*(x_{1,\lambda}) \quad (3-36)$$

类似地,对任意的  $x \in \Gamma_N$ , 下述结果也成立:

$$\begin{aligned} & q(x, u_N^*(0, x)) + J_N^*(x_N^*(1, x)) \\ & \leq q(x, u_N^*(0, x)) + J_{N-1}^*(x_N^*(1, x)) \\ & = J_N^*(x) \\ & \leq J_{N-1}^*(x) \end{aligned} \quad (3-37)$$

因此,可以选择

$$F_{tra}(x) = q(x, u_N^*(0, x)) + J_N^*(x_N^*(1, x)) \quad (3-38)$$

将式 (3-38) 中的  $x$  分别用  $x_1, x_{1,\lambda}$  代替,可以得到

$$\begin{aligned} & q(x_0, u_N^*(0, x_0)) + q(x_1, u_N^*(0, x_1)) + J_N^*(x_2) \\ & = q(x_0, u_N^*(0, x_0)) + F_{tra}(x_1) \\ & \leq q(x_0, u_{N,\lambda}^*(0, x_0)) + F_{tra}(x_{1,\lambda}) \\ & = q(x_0, u_{N,\lambda}^*(0, x_0)) + q(x_{1,\lambda}, u_N^*(0, x_{1,\lambda})) + J_N^*(x_N^*(1, x_{1,\lambda})) \end{aligned} \quad (3-39)$$

又由假设 3.1 可知,如果  $x_0 \in \Gamma_N$ , 则  $x_{1,\lambda}$  也属于  $\Gamma_N$ , 即  $x_{1,\lambda} \in \Gamma_N$ 。用  $x_{1,\lambda}$  替代不等式 (3-36) 中的  $x_0$ , 可以得到

$$\begin{aligned} & q(x_{1,\lambda}, u_N^*(0, x_{1,\lambda})) + J_N^*(x_N^*(1, x_{1,\lambda})) \\ & \leq q(x_{1,\lambda}, u_{N,\lambda}^*(0, x_{1,\lambda})) + J_N^*(x_{2,\lambda}) \end{aligned} \quad (3-40)$$

所以有

$$\begin{aligned} & q(x_0, u_N^*(0, x_0)) + q(x_1, u_N^*(0, x_1)) + J_N^*(x_2) \\ & \leq q(x_0, u_{N,\lambda}^*(0, x_0)) + q(x_{1,\lambda}, u_{N,\lambda}^*(0, x_{1,\lambda})) + J_N^*(x_{2,\lambda}) \\ & \leq q(x_0, u_{N,\lambda}^*(0, x_0)) + q(x_{1,\lambda}, u_{N,\lambda}^*(0, x_{1,\lambda})) + J_{N,\lambda}^*(x_{2,\lambda}) \end{aligned} \quad (3-41)$$

如此反复，我们可以得到

$$\begin{aligned} & q(x_0, u_N^*(0, x_0)) + \cdots + q(x_j, u_N^*(0, x_j)) + J_N^*(x_{j+1}) \\ & \leq q(x_0, u_{N,\lambda}^*(0, x_0)) + \cdots + q(x_{j,\lambda}, u_{N,\lambda}^*(0, x_{j,\lambda})) + J_{N,\lambda}^*(x_{j+1,\lambda}) \end{aligned} \quad (3-42)$$

因此，当  $j \rightarrow +\infty$  时，有

$$J_{RH}(x_0) \leq J_{RH,\lambda}(x_0)$$

证毕。

**注 3.3** 由定理 3.2 可知，按比例增大终端代价函数，采用预测控制所消耗的全局性能指标会增多。因此，在选择终端代价函数时，不能一味地追求大的终端状态约束集，而忽略终端代价函数对全局性能指标的影响。

## 3.5 求取一个合适的终端代价函数

由定理 3.1 和定理 3.2 可知，在选择终端代价函数时，要综合考虑吸引域与全局性能指标两个因素，依据侧重点的不同，选择不同的终端代价函数。至今，已有相关文献提出了一些求取终端代价函数的方法。文献[21]指出带比例参数的二次型函数可以用作预测控制的终端代价函数，并从理论上证明了通过选择合适的比例系数和二次型矩阵可以保证系统渐近稳定。文献[8]



通过将非线性系统在平衡点处进行一阶泰勒展开线性化，再利用最优控制理论得到了一个针对连续系统的二次型终端代价函数。文献[37]在终端状态集内抽取大量随机点，并求取这些随机点采用线性反馈控制所消耗的全局性能指标，最后采用支持向量机进行拟合计算，得到了一个终端代价函数，但这种方法不容易给出终端代价函数的显式表达式。

本节针对系统式(3-1)介绍一种简单的求取终端代价函数的方法，其基本思路是：首先将系统式(3-1)在原点处一阶泰勒展开线性化，利用线性最优控制理论求得一个初始的终端代价函数，再依据定理3.1、定理3.2及我们的需要对此初始终端代价函数进行调整得到最终的终端代价函数。

前面已经提到，理想的终端代价函数应为 $F(\cdot) = J_{\infty}^*(\cdot)$ ，而由于系统的非线性， $J_{\infty}^*(\cdot)$ 是无法得到的。但是，我们可以在原点处做到 $F(\cdot)$ 与 $J_{\infty}^*(\cdot)$ 尽量逼近。具体方法是：将系统式(3-1)围绕原点进行泰勒展开，取其一阶近似，得到系统式(3-1)的近似线性模型，再用线性最优控制理论求得与此 $J_{\infty}^*(\cdot)$ 相近的终端代价函数。文献[8]针对连续系统给出了具体的求解方法，此处，我们将文献[8]的方法拓展至离散系统。

将系统式(3-1)在原点处进行线性化处理，得到其近似线性模型如下：

$$x_{k+1} = Ax_k + Bu_k \quad (3-43)$$

其中

$$A = (\partial f / \partial x)(0,0)$$

$$B = (\partial f / \partial u)(0,0)$$

假设此近似线性系统是稳定的，则可以用线性最优控制理论得到一个初始终端代价函数，具体步骤如下。

**第一步：**求解黎卡堤方程得到 $G_0$ 。

$$\mathbf{G}_0 = \mathbf{A}^T \mathbf{G}_0 \mathbf{A} - (\mathbf{A}^T \mathbf{G}_0 \mathbf{B})(\mathbf{B}^T \mathbf{G}_0 \mathbf{B} + \mathbf{R})^{-1} (\mathbf{B}^T \mathbf{G}_0 \mathbf{A}) + \mathbf{Q} \quad (3-44)$$

第二步：求得系统的一个局部线性稳定反馈增益  $\mathbf{K}$ 。

$$\mathbf{K} = -(\mathbf{B}^T \mathbf{G}_0 \mathbf{B} + \mathbf{R})^{-1} (\mathbf{B}^T \mathbf{G}_0 \mathbf{A}) \quad (3-45)$$

第三步：求解黎卡堤方程得到  $\mathbf{G}_K$ 。

$$(\alpha \mathbf{A}_K)^T \mathbf{G}_K (\alpha \mathbf{A}_K) - \mathbf{G}_K = -\mathbf{Q}_K \quad (3-46)$$

其中

$$\mathbf{A}_K = \mathbf{A} + \mathbf{B}\mathbf{K}$$

$$\mathbf{Q}_K = \mathbf{Q} + \mathbf{K}^T \mathbf{R} \mathbf{K}$$

$\alpha \in [1, +\infty)$  是一个可调参数，满足

$$\alpha |V_{\max}(\mathbf{A}_K)| < 1$$

其中， $V_{\max}(\mathbf{A}_K)$  为  $\mathbf{A}_K$  的幅值最大的特征值。

通过上述步骤得到  $\mathbf{G}_K$  后，便可将  $F(x) = x^T \mathbf{G}_K x$  用作初始终端代价函数。然后以定理 3.1 和定理 3.2 为基础，依据我们的需要对  $F(x)$  进行调整，以得到一个合适的终端代价函数， $\lambda F(x) = x^T (\lambda \mathbf{G}_K) x$ ， $\lambda > 0$ 。例如，如果要求终端状态约束集大一些，而对预测控制消耗的全局性能指标并不是很关心，则可以选择一个较大的  $\lambda$ ；相反，如果要求消耗的全局性能指标尽量少，而系统的吸引域相较于其工作区域要大得多，则可以选择一个较小的  $\lambda$ 。

## 3.6 具有可变自调节终端代价函数的 NMPC 方案

在 3.5 节中，我们讨论了终端代价函数与终端状态集及全局性能指标之间的关系。由定理 3.1 和定理 3.2 可知，终端代价函数的增大不仅能增大终

端状态集,同时也增加了全局性能指标的消耗。因此,在预测控制选取终端代价函数时,应综合考虑对二者的影响。本节提出了一种具有自调节终端代价函数的模型预测控制方案,使终端代价函数能随着控制过程的推进而进行适当的调整,可用较少的全局性能指标消耗保证闭环系统的性能。

### 3.6.1 问题描述

考虑非线性离散系统式(3-1)

$$x(k+1) = f(x(k), u(k))$$

其中,  $x \in R^n$  为系统状态,  $u \in R^m$  为系统输入,  $k = 0, 1, \dots$  为采样时刻,  $f(\cdot, \cdot)$  为非线性函数, 且有  $x(0) = x_0, f(x_0, 0) = x_0$ 。系统须服从状态约束和输入约束  $(x, u) \in Z, Z \in X \times U$ , 满足  $Z, X, U$  为紧的。 $Z$  在  $X$  上的投影表示为  $Z_X$ 。

本节要考虑的模型预测控制方案是在一般模型预测控制问题的基础上, 引入可变自调节因子  $\beta(k)$ , 从而使终端代价函数具有了可变自调节性能。具有自调节终端代价函数的 NMPC 方案的最优化问题可描述为

$$\min_u J(x, u, \beta) = \sum_{i=0}^{N-1} q(x(i|k), u(i|k)) + \beta(k)q(x(N|k), u(N|k)) \quad (3-47)$$

满足以下约束条件:

$$\begin{aligned} x(0|k) &= x \\ x(i+1|k) &= f(x(i|k), u(i|k)), i \in I_{[0, N-1]} \\ (x(i|k), u(i|k)) &\in Z, i \in I_{[0, N]} \\ x(N|k) &= f(x(N|k), u(N|k)) \\ q(x(N|k), u(N|k)) &\leq \lambda(k) \end{aligned} \quad (3-48)$$

式(3-47)中的  $\beta(k)q(x(N|k), u(N|k))$  即本节所指的可变自调节终端代价函数, 表示为  $Q(N|k)$ , 其可变自调节性能由自调节因子  $\beta(k)$  来实现。

式 (3-48) 中包含了广义终端约束条件, 意味着预测终端状态  $x(N|k)$  必须对任意稳态可达。优化问题式 (3-47) 的最优输入序列为

$$u^* := [u^*(0|k), \dots, u^*(N|k)]$$

相应的状态序列为

$$x^* := [x^*(0|k), \dots, x^*(N|k)]$$

在预测控制中, 通常仅将  $k$  时刻最优输入序列的第一项  $u^*(0|k)$  代入系统中。不难看出, 终端代价函数值与终端代价项自调节因子  $\beta$  的取值直接相关。式 (3-48) 中的参数序列  $\lambda(k)$  会根据先验终端状态代价进行更新, 则  $k+1$  时刻的闭环系统可描述为

$$\begin{aligned} x(k+1) &= f(x(k), u^*(0|k)) \\ \lambda(k+1) &= q(x^*(N|k), u^*(N|k)) \end{aligned} \quad (3-49)$$

式 (3-48) 和式 (3-49) 中的序列  $\lambda(k)$  为非增且有下界 ( $q(x_s, u_s) = 0$ ) 的收敛序列。用  $\lambda_\infty := \lim_{k \rightarrow \infty} \lambda(k) \geq 0$  定义其极限值, 则  $\lambda_\infty$  是闭环系统预测终端状态极限值。

下面通过一个简单实例证明, 当采用本节的模型预测控制方案时,  $\beta$  取值的大小对  $\lambda_\infty$  值并无决定性影响, 可以选择较小的终端代价函数值使预测的终端稳态代价任意地接近最优可达稳态代价, 也就是说, 可用较少的全局性能指标消耗保证闭环系统的性能。

考虑有状态约束和输入约束的离散时间非线性系统

$$\begin{aligned} x(k+1) &= x(k)[u(k)-1] \\ Z &= X \times U \\ U &= [-3.5, 3.5] \\ X &= [-12, 12] \end{aligned}$$

代价函数

$$q(x,u) = \frac{2}{5}x^4 - \frac{12}{5}x^3 + \frac{3}{5}x^2 + 35 + (u-3)^2$$

预测域  $N=1$ 。

采用本节的 NMPC 方案，仿真结果如图 3-1 所示。图 3-1 表明了  $\beta$  取 4 个不同常值时的闭环状态序列。可以看出，当  $\beta=2.5$  和  $\beta=10$  时， $x$  收敛于 0，因此， $\lambda_\infty = q(0,3) = 35$ ；当  $\beta=5$  和  $\beta=12.5$  时， $x$  收敛于 5，因此， $\lambda_\infty = q(5,3) = 0$ 。由本例结果可以看出， $\beta$  取值的大小对  $\lambda_\infty$  值并无决定性影响，可以选取较小的终端代价函数值使预测的终端稳态代价任意地接近最优可达稳态代价，即可用较少的全局性能指标消耗保证闭环系统的性能。

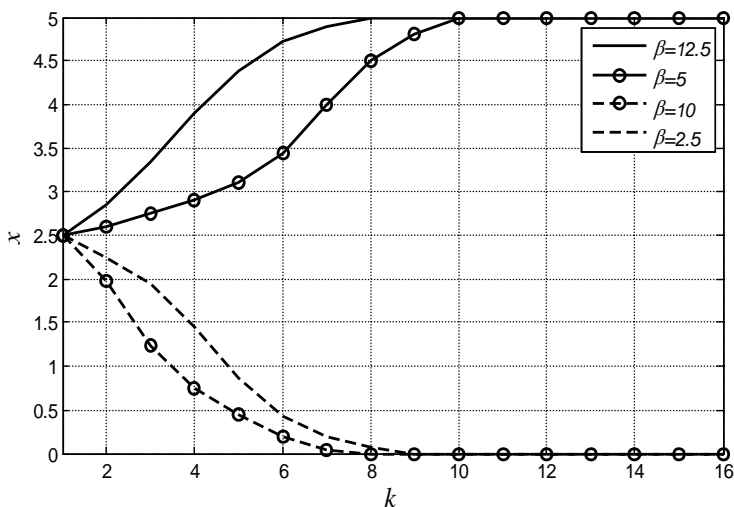


图 3-1  $\beta$  取不同值时的闭环系统状态序列

### 3.6.2 对闭环系统性能的证明

本节将研究式 (3-47) 中的自调节因子  $\beta$  可变，从而引起终端代价函数变化时，闭环系统式 (3-49) 的渐近稳定性。也就是说，当自调节因子  $\beta$  根

据某适当选取的自调节规则  $B$  变化时, 闭环系统的性能指标函数必须有界, 这样才能保证增加终端代价函数后的 NMPC 优化问题可近似为无穷时域最优优化问题, 从而保证系统的渐近稳定性。自调节规则  $B$  的表示形式如下:

$$\begin{aligned}\beta(k+1) &= B(\beta(k), x(k), \lambda(k)) \\ \beta(0) &= \beta_0 \geq 0\end{aligned}\quad (3-50)$$

为了证明闭环系统性能指标函数的有界性, 我们给出以下关于  $\beta$  的假设。

**假设 3.3** 令  $e(k) := \beta(k+1) - \beta(k)$ 。对所有的  $k \in I_{\geq 0}$  和某些约束  $c, \underline{\beta} \in R$ , 式 (3-28) 的自调节规则  $B$  可以保证序列  $\beta(\cdot)$  满足  $e(k) \leq c, \beta(k) \geq \underline{\beta}$ , 且有

$$\begin{aligned}\lim_{k \rightarrow \infty} e(k) &< 0 \\ \lim_{k \rightarrow \infty} \beta(k) &< \infty\end{aligned}$$

**定理 3.3** 假设式 (3-47) 的最优化问题在  $k=0$  处有可行解。若闭环系统式 (3-49) 和式 (3-50) 满足假设 3.3, 则有

$$\lim_{T \rightarrow \infty} \frac{1}{T} \sum_{k=0}^{T-1} q(x(k), u(k)) \leq \lambda_{\infty} \quad (3-51)$$

证明:

在一般的模型预测控制问题中, 问题式 (3-47) 在  $k+1$  时刻的一个可行解是由先前的最优解末项加上稳态输入给出的, 即

$$\hat{u}(k+1) := [u^*(1|k), \dots, u^*(N|k), u^*(N|k)]$$

由此可得

$$\begin{aligned}& J(x(k+1), u^*(k+1), \beta(k+1)) - J(x(k), u^*(k), \beta(k)) \\ & \leq J(x(k+1), \hat{u}(k+1), \beta(k+1)) - J(x(k), u^*(k), \beta(k)) \\ & = (1 + e(k))q(x^*(N|k), u^*(N|k)) - q(x(k), u(k))\end{aligned}\quad (3-52)$$

如前所述, 序列  $q(x^*(N|k), u^*(N|k))$  在  $k$  时刻是非增的, 且当  $k \rightarrow \infty$  时收敛于  $\lambda_\infty$ 。这表明当  $k \rightarrow \infty$  时,  $\varepsilon(k) := q(x^*(N|k), u^*(N|k)) - \lambda_\infty$  收敛于 0。另外, 对所有  $k \in I_{\geq 0}$ , 有

$$0 \leq \varepsilon(k) \leq \varepsilon(0) < \infty$$

对任一  $T \geq 1$ , 由不等式 (3-52) 可得

$$\begin{aligned} & J(x(T), u^*(T), \beta(T)) - J(x_0, u_0^*, \beta_0) \\ & \leq \sum_{k=0}^{T-1} [(1+e(k))(\lambda_\infty + \varepsilon(k)) - q(x(k), u(k))] \end{aligned} \quad (3-53)$$

首先, 考虑满足假设 3.3 中  $\lim_{k \rightarrow \infty} e(k) < 0$  的情况。对式 (3-53) 两边都取极限平均值, 得到

$$\begin{aligned} & \lim_{T \rightarrow \infty} (1/T) (J(x(T), u^*(T), \beta(T)) - J(x_0, u_0^*, \beta_0)) \\ & \leq \lambda_\infty + \lim_{T \rightarrow \infty} (1/T) \left( \sum_{k=0}^{T-1} -q(x(k), u(k)) \right) + \\ & \quad \lim_{T \rightarrow \infty} (1/T) \left( \sum_{k=0}^{T-1} [e(k)\lambda_\infty + (1+e(k))\varepsilon(k)] \right) \\ & \leq \lambda_\infty - \lim_{T \rightarrow \infty} (1/T) \left( \sum_{k=0}^{T-1} q(x(k), u(k)) \right) \end{aligned} \quad (3-54)$$

这里最后一个不等式遵循以下事实:

$$\begin{aligned} \lim_{k \rightarrow \infty} \varepsilon(k) &= 0 \\ \lim_{k \rightarrow \infty} e(k) &\leq 0 \\ \lambda_\infty &\geq 0 \end{aligned}$$

且对于所有  $k \in I_{\geq 0}$ , 有

$$\begin{aligned} 0 &\leq \varepsilon(k) \leq \varepsilon(0) < \infty \\ e(k) &\leq c \leq \infty \end{aligned}$$

另外, 若对于所有  $k \in I_{\geq 0}$  都有  $\beta(k) \geq \underline{\beta}$ , 且满足  $q$  为连续的、 $Z$  为紧的,

则通过定义式 (3-47) 中的代价函数  $J$  可得, 存在一个常量  $\underline{J} \in R$ , 满足对所有  $k \in I_{\geq 0}$ , 有

$$J(x(k), u^*(k), \beta(k)) \geq \underline{J}$$

因此可得

$$\begin{aligned} & \lim_{T \rightarrow \infty} \frac{J(x(T), u^*(T), \beta(T)) - J(x_0, u_0^*, \beta_0)}{T} \\ & \geq \lim_{T \rightarrow \infty} (\underline{J} / T) + \lim_{T \rightarrow \infty} \frac{-J(x(0), u_0^*, \beta_0)}{T} \\ & = 0 \end{aligned} \quad (3-55)$$

结合式 (3-54) 和式 (3-55) 得出式 (3-51) 是成立的。

其次, 考虑满足假设 3.3 中  $\lim_{k \rightarrow \infty} \beta(k) < \infty$  的情况。此时, 存在一个无限的时间点序列  $\{k_i\} \in I_{\geq 0}$ , 满足对所有的  $k_i$  和某些  $\beta_\infty < \infty$ , 有  $\beta(k_i) \leq \beta_\infty$ 。由  $e(k)$  的定义得

$$\sum_{k=0}^{k_i-1} e(k) = \beta(k_i) - \beta_0$$

由假设可得对所有的  $k \in I_{\geq 0}$  有  $e(k) \leq c$ , 由式 (3-53) 可得

$$\begin{aligned} & J(x(k_i), u^*(k_i), \beta(k_i)) - J(x_0, u_0^*, \beta_0) \\ & \leq (k_i + \beta_\infty - \beta_0) \lambda_\infty + \sum_{k=0}^{k_i-1} [(1+c)\varepsilon(k) - q(x(k), u(k))] \end{aligned} \quad (3-56)$$

不等式两边取极限平均值可得

$$\begin{aligned} & \lim_{T \rightarrow \infty} (1/T) (J(x(T), u^*(T), \beta(T)) - J(x_0, u_0^*, \beta_0)) \\ & \leq \lim_{i \rightarrow \infty} (1/k_i) (J(x(k_i), u^*(k_i), \beta(k_i)) - J(x_0, u_0^*, \beta_0)) \\ & \leq \lim_{i \rightarrow \infty} (1/k_i) \left( (k_i + \beta_\infty - \beta_0) \lambda_\infty + \sum_{k=0}^{k_i-1} [(1+c)\varepsilon(k) - q(x(k), u(k))] \right) \\ & \leq \lambda_\infty + \lim_{i \rightarrow \infty} (1/k_i) \sum_{k=0}^{k_i-1} [(1+c)\varepsilon(k) - q(x(k), u(k))] \end{aligned} \quad (3-57)$$



$$\begin{aligned}
 &\leq \lambda_{\infty} - \lim_{i \rightarrow \infty} (1/k_i) \sum_{k=0}^{k_i-1} q(x(k), u(k)) \\
 &\leq \lambda_{\infty} - \lim_{T \rightarrow \infty} (1/T) \sum_{k=0}^{T-1} q(x(k), u(k))
 \end{aligned} \tag{3-57 续}$$

这里第四个不等式再次由“对于所有  $k \in I_{\geq 0}$  有  $\lim_{k \rightarrow \infty} \varepsilon(k) = 0$  和  $\varepsilon(k) \leq \varepsilon(0) < \infty$ ”这一事实得到, 且最后一个不等式遵循“ $\{k_i\}$  是  $I_{\geq 0}$  的一个无限子序列”这一事实。由式 (3-57) 也可得出式 (3-51) 是成立的。至此, 便证明了带有可变自调节终端代价项的闭环系统性能指标函数的有界性, 则闭环系统 (3-49) 是渐近稳定的。

### 3.6.3 对自调节规则的说明

在 3.6.2 节中已经证明了预测终端状态极限值  $\lambda_{\infty}$  为闭环系统性能指标函数的上界, 在本小节中将给出使  $\beta$  满足假设 3.3 且使  $\lambda_{\infty}$  取较小值的自调节规则  $B$ 。显然, 由文献[118]中定义的  $q(x_s, u_s)$  可知, 对每一种自调节规则  $B$  都有  $\lambda_{\infty} \geq q(x_s, u_s) = 0$ 。一般说来, 在不对系统和代价函数做进一步假设的前提下, 期望的最好结果是  $\lambda_{\infty}$  与相应的自调节规则的  $\omega$ -极限集合的最优可达稳态代价相等。在本小节中, 我们将给出自调节规则的一个充分条件(定理 3.4), 使上述性质成立。

令  $\omega_B(x_0)$  为闭环状态序列 (3-49) 的  $\omega$ -极限集合, 初始状态为  $x_0$ , 自调节规则  $B$  的形式如式 (3-50) 所示。给出如下的定理。

**定理 3.4** 假设自调节规则  $B$  对所有序列  $x(\cdot)$  和  $\lambda(\cdot)$  都成立, 且当  $x(\cdot)$  和  $\lambda(\cdot)$  作为式 (3-50) 的开环输入量时, 使下式成立:

$$\lambda_{\infty} - \lim_{k \rightarrow \infty} q_{\min}(x(k)) > 0 \Rightarrow \lim_{k \rightarrow \infty} \beta(k) \rightarrow \infty \tag{3-58}$$

对闭环系统式 (3-49) 和式 (3-50), 极限  $\lim_{k \rightarrow \infty} q_{\min}(x(k))$  存在, 且满足

$$\lambda_{\infty} = \lim_{k \rightarrow \infty} q_{\min}(x(k)) \leq \bar{q}_{\min}(y), \quad y \in \omega_B(x_0) \tag{3-59}$$

证明（用反证法）：

首先，假设如果  $\beta$  满足式（3-58），则

$$\lambda_{\infty} - \lim_{k \rightarrow \infty} q_{\min}(x(k)) \leq 0$$

即假设  $\lambda_{\infty} - \lim_{k \rightarrow \infty} q_{\min}(x(k)) > 0$  的矛盾结果成立。由式（3-58）可得

$$\lim_{k \rightarrow \infty} \beta(k) \rightarrow \infty$$

但是，文献[118]已经证明，若对任一  $\varepsilon > 0$  都有  $\beta > a / \varepsilon$ （ $a$  是有限常数，且  $a > 0$ ），那么对所有的  $x(k)$  都有

$$q(x^*(N|k), u^*(N|k)) \leq q_{\min}(x(k)) + \varepsilon$$

然而，由  $\lim_{k \rightarrow \infty} \beta(k) \rightarrow \infty$  可得

$$\lambda_{\infty} = \lim_{k \rightarrow \infty} q_{\min}(x(k))$$

这与开始的假设矛盾。由  $\lambda_{\infty}$  和  $q_{\min}$  的定义可知，极限  $\lim_{k \rightarrow \infty} q_{\min}(x(k))$  存在且有

$$\lambda_{\infty} - \lim_{k \rightarrow \infty} q_{\min}(x(k)) = 0$$

这证明了式（3-59）中的等号是成立的。

其次，根据  $\omega$ -极限集合  $\omega_B(x_0)$  的定义，对任一  $y \in \omega_B(x_0)$  和任一  $\varepsilon > 0$  存在一个无限的时间序列  $\{k_i^y\}$ ，使  $x(k_i^y) \in B_{\varepsilon}(y) \cap Z_X$  成立，同时表明

$$\lambda_{\infty} = \lim_{k \rightarrow \infty} q_{\min}(x(k)) = \lim_{i \rightarrow \infty} q_{\min}(x(k_i^y)) \leq q_{\min}(y, \varepsilon)$$

当其任一  $y \in \omega_B(x_0)$  和任一  $\varepsilon > 0$  都成立时，得到

$$\lambda_{\infty} = \lim_{k \rightarrow \infty} q_{\min}(x(k)) \leq \bar{q}_{\min}(y), \quad y \in \omega_B(x_0)$$

证毕。

在定理 3.4 中, 能证明最优可达稳态代价沿闭环系统收敛, 即极限  $\lim_{k \rightarrow \infty} q_{\min}(x(k))$  是存在的。而  $\lambda_{\infty}$  值与此极限值相等, 那么说明它至少与闭环系统在  $\omega$ -极限集合的最优鲁棒可达稳态代价相等, 这对于自调节规则来说是期望的理想条件, 因此, 定理 3.4 可视为自调节规则的一个充分条件。

### 3.6.4 几种自调节规则及其性质

在本小节中, 我们将针对自调节因子  $\beta$  提出几种自调节规则, 并确保其满足定理 3.3 和定理 3.4 中规定的条件。首先, 为了使  $\beta$  单调递增, 设定了最简单的自调节规则 1。其次, 引入参数  $\sigma(k) := q(x^*(N|k), u^*(N|k)) - q_{\min}(x(k))$ , 设定了较为复杂的自调节规则 2 和自调节规则 3。

**自调节规则 1:**  $B_1(\beta(k), x(k), \lambda(k)) := \beta(k) + \varepsilon$ , 其中  $\varepsilon > 0$ 。

**自调节规则 2:**  $B_2(\beta(k), x(k), \lambda(k)) := \beta(k) + \alpha(\sigma(k))$ , 其中  $\alpha \in \mathcal{K}$ 。

**自调节规则 3:**  $B_3(\beta(k), x(k), \lambda(k)) := \begin{cases} 1, & C(k) \leq 0 \\ \beta(k) + \eta(\sigma(k)), & \text{其他} \end{cases}$ , 其中

$C(0) = 0$ , 且对每个  $k \in I_{\geq 1}$ , 有  $C(k) := q(x^*(N|k), u^*(N|k)) - q(x^*(N|k_l), u^*(N|k_l)) + \nu(\sigma(k))$ , 式中  $k_l := \max_{s \leq k, \beta(s)=1} s - 1$ ,  $\nu, \eta \in \mathcal{K}$ 。

由上述自调节规则的定义式可以看出, 自调节规则 1 与自调节规则 2 相比的一个最大的优点是, 不需要在每个时刻  $k$  都知道  $q_{\min}(x(k))$  的值。但是, 自调节规则 1 并不是在  $\beta \rightarrow \infty$  的任意情形下都是可取的。与自调节规则 1 和自调节规则 2 相比, 在自调节规则 3 下, 自调节因子  $\beta$  可以根据需要随时重新设定, 因此能更好地满足闭环系统的性能要求。

下面通过一个实例对自调节规则的性质进行分析。由于自调节规则 1 的性质是显而易见的, 因此本小节的实例仅对自调节规则 2 和自调节规则 3 的性质进行分析。

考虑如下的离散时间非线性系统：

$$x(k+1) = x(k) - x(k)u(k)$$

状态约束和输入约束为

$$Z = X \times U = [0,1]^2$$

代价函数为

$$q(x, u) = x + u^2 / 2$$

预测域  $N=1$ 。最优稳态为  $(x_s, u_s) = (0, 0)$ ，且对所有的  $x \in X$ ，有  $q_{\min}(x) = q(x_s, u_s) = 0$ 。  $u^*(0|k) = \min\{\beta(k)x(k), 1\}$ ， $u^*(1|k) = 0$ ，则最优化问题式 (3-47) 和式 (3-48) 的最优解很容易得出。 $\beta(k)$  的值根据不同的自调节规则得出。

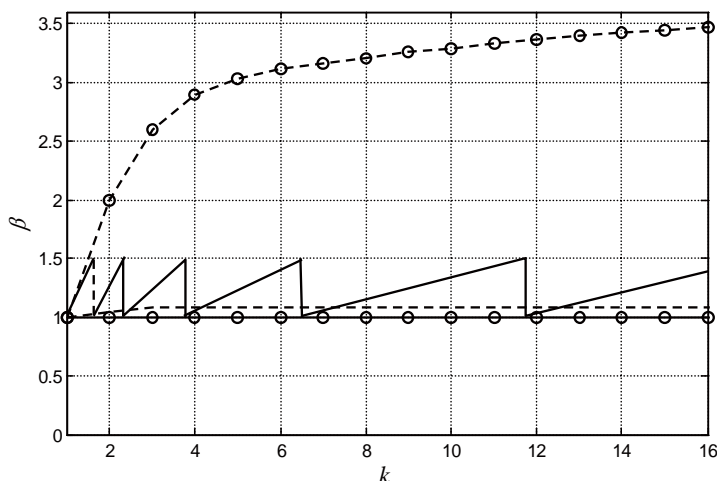
闭环系统可以表示为

$$x(k+1) = x(k) - x(k)\min\{\beta(k)x(k), 1\}$$

对上述的任一自调节规则和任一初始状态  $x_0 \in X$  和  $\beta_0 > 0$ ， $\omega$ -极限集合都是由  $\omega_B(x_0) = \{0\}$  给出的，且有

$$\max_{y \in \omega_B(x_0)} \bar{q}_{\min}(y) = \min_{y \in \omega_B(x_0)} \bar{q}_{\min}(y) = 0$$

在自调节规则 1 下，由定义式可以直接看出， $\beta$  总是无界的。在自调节规则 2 下，当  $\alpha(r) = r$  时， $x$  会缓慢地收敛到零，此时  $\beta$  也不会保持有界，如图 3-2 中的带圈虚线所示；当  $\alpha(r) = r^2$  时， $\beta$  会保持有界且收敛，如图 3-2 中的虚线所示。在自调节规则 3 下，当  $\nu(r) = r^2, \eta(r) = r$  时， $\beta \equiv 1$ ，如图 3-2 中的带圈实线所示；当  $\nu(r) = \eta(r) = r$  时， $\beta$  的值呈锯齿状变化，有界但不收敛，如图 3-2 中的实线所示。由本例的仿真图 3-2 可以看出，当采用不同的自调节规则时， $\beta$  值在闭环系统中的变化规律是不同的。


 图 3-2  $\beta$  在不同自调节规则下的变化曲线

### 3.7 仿真算例

本节通过仿真验证 3.3 和 3.4 节得到的关于终端代价函数的两条性质，并说明如何采用 3.5 节的方法求取预测控制的终端代价函数。

同样以第 2 章中的例子为本次仿真的对象，即

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T \\ T & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} T\mu \\ T\mu \end{bmatrix} u_k + \begin{bmatrix} T(1-\mu) & 0 \\ 0 & -4T(1-\mu) \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} u_k$$

其中， $\mu=0.5$ ；状态约束与输入约束分别为  $X=\{x|\|x\|_1\leq 4\}$ ， $U=\{u|\|u\|\leq 2\}$ ；采样时间  $T=0.1s$ 。阶段指标函数为二次型性能指标， $q(x,u)=x^T Q x + u^T R u$ ，其中， $Q=T\times 0.5I$ ， $R=T\times 1$ 。

首先将仿真模型在原点处进行泰勒一阶展开处理为线性近似系统

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0.1 \\ 0.1 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 0.05 \\ 0.05 \end{bmatrix} u_k$$

然后利用线性最优控制理论求得此线性系统的局部线性稳定反馈增益

$$K = -[2.0107 \ 2.0107]$$

并求得  $A_K$  的幅值最大的特征值的幅值为

$$|V_{\max}(A_K)| = 0.9000$$

取可调参数  $\alpha = 1.11$ ，即可得到一个初始终端代价函数为  $F(x) = x^T G x$ ，其中

$$G = [1107.356 \ 857.231; 857.231 \ 1107.356]$$

假设需要终端状态约束集大一些，则可以增大初始终端代价函数。此处，我们将其增大 10 倍，即  $10F(x) = x^T (10G)x$  为增大后的终端代价函数。为方便起见，用 (A) 表示终端代价函数为  $F(x) = x^T G x$  的预测控制，用 (B) 表示终端代价函数为  $10F(x) = x^T (10G)x$  的预测控制，采用第 2 章中求取终端状态约束集的方法，可以得到 (A) 和 (B) 的终端状态约束集，如图 3-3 所示。

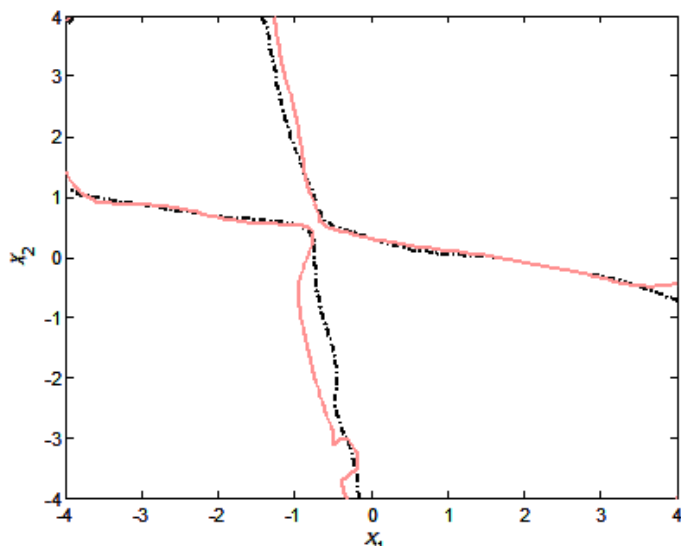


图 3-3 终端状态约束集对比示意图

在图 3-3 中, 点画线围起来的区域为 (A) 的终端状态约束集, 实线围起来的区域为 (B) 的终端状态约束集。可以看出, (B) 的终端状态约束集比 (A) 的大一些。

但是, 正如定理 3.2 所示, 按比例增大终端代价函数, 采用预测控制所消耗的全局性能指标会增多。为验证此结论, 取预测步长为  $N=3$ , 并在 (A) 的吸引域中选取任意点, 分别用 (A) 和 (B) 进行控制。其收敛曲线和消耗的性能指标分别如图 3-4 和表 3-1 所示。

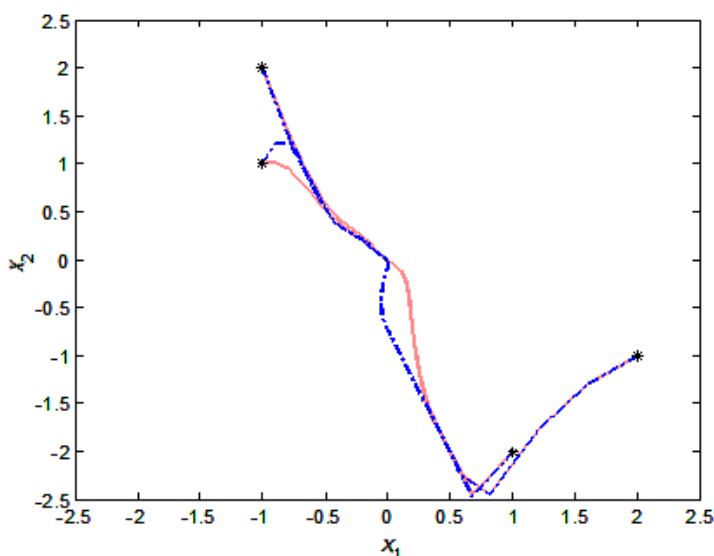


图 3-4 状态收敛轨迹对比

表 3-1 全局性能指标对比

初始状态点	(2,-1)	(1,-2)	(-1,2)	(-1,1)
(A) 的全局性能指标	34.7915	18.2460	16.4528	7.9936
(B) 的全局性能指标	37.0120	20.6714	23.2854	23.8467

在图 3-4 中, 实线表示 (A) 的状态收敛曲线, 点画线表示 (B) 的状态

收敛曲线。可以看出，在相同的预测步长下，(A)的吸引域中的点都可由(B)引导至原点。但是，由表 3-1 可以看出，(B)所消耗的全局性能指标却要比(A)大。

下面采用 3.6 节中的方法对上述算例中的系统进行研究，以便与其他方法进行比较。采用 3.6.4 节中的自调节规则 2 对终端代价函数进行调节，即令  $\tilde{F}(x) = B_2(\cdot)F(x) = (\beta(k) + \alpha(\sigma(k)))F(x)$ 。

为了便于比较，用 (a) 表示采用放大后的终端代价函数  $\bar{F}(x) = 10F(x) = x^T(10G)x$  的预测控制，用 (b) 表示采用原始终端代价函数  $F(x) = x^TGx$  的预测控制，用 (c) 表示采用 3.6 节自调节终端代价函数  $\tilde{F}(x) = (\beta(k) + \alpha(\sigma(k)))F(x)$  的预测控制，令  $\alpha(r) = r^2$ 。采用第 2 章中介绍的求取终端状态约束集的方法，可分别得到 (a)、(b) 和 (c) 的终端状态约束集，如图 3-5 所示。

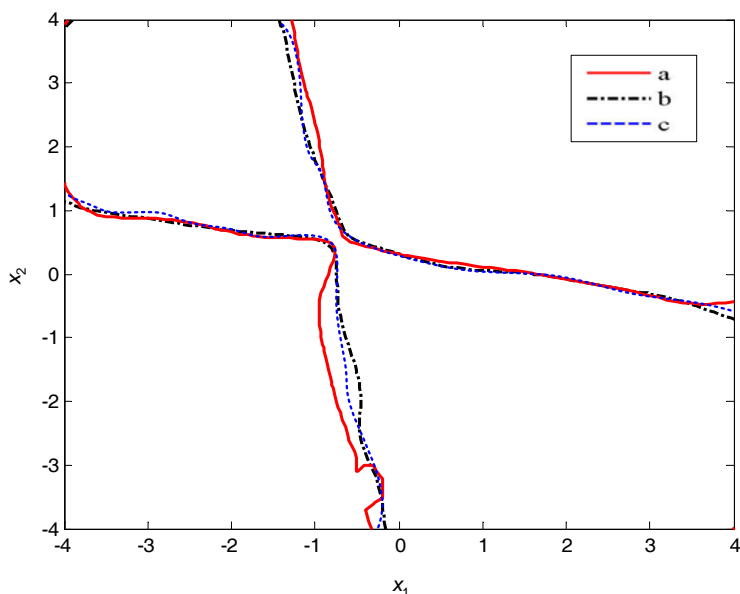


图 3-5 终端状态约束集对比示意图



在图 3-5 中, 实线之间的区域代表 (a) 的终端状态约束集, 点画线之间的区域代表 (b) 的终端状态约束集, 虚线之间的区域代表 (c) 的终端状态约束集。可以看出, (a) 的终端状态约束集最大, 而 (b) 和 (c) 的终端状态约束集大小并无明显差距。可见, 方案 (c) 与原始方案相比并未引起终端状态集的明显缩小, 因此并未使系统的吸引域减小。接下来再对比采用三种方案时的全局性能指标消耗。由于定理 3.2 已经证明, 若按比例增大终端代价函数, 则预测控制过程对全局性能指标的消耗必然会增多。因此, 仅对 (b) 和 (c) 的全局性能指标消耗进行对比。

取预测步长  $N=4$ , 并在 (b) 和 (c) 共同的吸引域中选取几个不同的点, 分别用 (b) 和 (c) 的方案进行控制, 各自消耗的性能指标如表 3-2 所示。由表中结果可得, 在相同的预测步长下, 将选取的点成功引导至原点后, (c) 所消耗的全局性能指标要小于 (b)。可见, 采用 3.6 节的方法可在不减小系统吸引域的前提下, 降低全局性能指标的消耗。

表 3-2 全局性能指标消耗对比

初始状态点	(3,-2)	(3,-3)	(-2,2)	(-2,3)
(b)	32.65	31.72	14.57	9.73
(c)	31.46	29.48	13.19	8.36

### 3.8 本章小结

关于终端代价函数与终端状态约束集及全局性能指标之间的关系, 现有相关研究比较少见。本章首先分析了预测控制的终端代价函数与终端状态约束集以及终端代价函数与全局性能指标之间的定性关系, 证明了按比例增大终端代价函数会增大终端状态约束集, 但同时也会使预测控制的全局性能指标消耗得更多; 其次, 在此基础上给出了一种简单的求取终端代价函数的方

法，此终端代价函数是通过将非线性系统在原点处线性化展开，利用线性最优控制理论得到的；再次，以终端代价函数与终端状态约束集以及终端代价函数与全局性能指标之间的定性关系为准则，对利用线性最优控制理论得到的终端代价函数进行比例调整，得到最终的终端代价函数；最后，提出了一种具有自调节终端代价函数的模型预测控制方案，使终端代价函数能随着控制过程的推进而进行适当的调整，可用较少的全局性能指标消耗保证闭环系统的性能。

## 第 4 章

# 鲁棒预测控制方法

### 4.1 引言

在第 1 章中已经提到, 预测控制器本身具有一定的鲁棒性<sup>[30]</sup>。但是当系统的实际模型与其标称模型差别较大时, 系统的稳定性得不到保证, 需要设计具有强鲁棒性的预测控制算法。目前, 对于鲁棒预测控制算法的设计按照系统模型的不同主要分为三类: 一类基于多胞模型; 一类基于逼近模型 (如神经网络模型、模糊模型等); 一类基于扰动模型。其中, 基于扰动模型的鲁棒预测控制因为物理意义直观明了、适用范围大等优点吸引了众多的眼球。文献[41,42]针对受扰系统分别设计了滚动时域  $H_\infty$  预测控制器; 文献[43]针对多胞不确定系统设计了基于椭圆不变集的鲁棒预测控制器; 文献[44]用多面体不变集替代了文献[43]中的椭圆不变集, 扩大了系统的初始可行域; 文献[45,46]设计了基于近似预测可达集的预测控制器; 文献[47]针对有界扰动系统给出了一个通用的 min-max MPC 控制器; 文献[48]以文献[47]为基础, 先离线计算出一个线性反馈控制增益, 将待优化控制律看成一个仿射输入控制律 (线性反馈控制律与修正量的和), 再对修正量进行在线优化求解得到最终控制律。

本章以基于扰动模型的鲁棒预测控制为研究对象，首先给出系统的干扰模型，然后设计了三种基于扰动模型的鲁棒预测控制算法。

考虑如下有界干扰离散模型：

$$x_{k+1} = f(x_k, u_k) + w_k \quad (4-1)$$

其中， $x_k \in R^n, u_k \in R^m$  分别为系统在采样时刻  $k$  的状态和输入； $w_k \in R^w$  为系统干扰；非线性映射  $f$  满足  $f(0,0)=0$ 。系统的状态约束和输入约束分别为  $x_k \in X, u_k \in U$ ， $X$  和  $U$  都是紧凸的，且都包含原点，系统干扰  $w_k \in W$  不随时间衰减至零，但满足 2-范数有界，即  $\|w_k\|_2 \leq \bar{W}$ ， $\bar{W}$  为其 2-范数的上界。

考虑如下两条假设。

**假设 4.1** 系统状态已知且准确。

**假设 4.2** 存在这样一个原点邻域  $\Omega \subseteq X$ ，其满足鲁棒控制不变集条件 (RCIS 条件)：对  $\forall x \in \Omega$ ，存在  $u \in U$ ，使得对  $\forall w \in W$ ，都有  $f(x, u) + w \in \Omega$ 。

$\Omega$  称为鲁棒控制不变集。如果没有外来干扰或干扰随时间逐渐衰减至零，则系统状态可被稳定在原点。但是对于系统式 (4-1) 来说，系统状态是不可能被稳定在原点的，只能被控制在某个原点邻域内。

## 4.2 以集合包含度为优化指标的鲁棒预测控制

受文献[45,46]的启发，本节设计了一个基于近似预测可达集的鲁棒预测控制器。为避免双模控制中控制量的切换问题，以终端状态约束集对可达集的包含度为优化指标，并采用预测步长逐渐减小的控制策略，将吸引域内的状态点在有限步长内引导至终端状态约束集，当状态点进入终端状态约束集之后，仍采用预测控制使其始终停留在此集之内。

### 4.2.1 预测可达集与集合包含度的定义

为方便起见, 首先定义关于集合的相关运算。考虑集合  $A$ , 将集合运算  $g(A)$  定义为  $g(A) = \{g(x) | x \in A\}$ 。

$k$  时刻, 已知状态为  $x_k$ , 考虑控制序列

$$u(x_k) = \{u(0, x_k), u(1, x_k), u(2, x_k), \dots\}$$

系统在  $k+i$  ( $i \geq 1$ ) 时刻的预测可达集可描述为

$$X_i(x_k, u(x_k)) = f(X_{i-1}(x_k, u(x_k)), u(i-1, x_k)) + W \quad (4-2)$$

其中

$$X_0(x_k, u(x_k)) = x_k$$

考虑集合  $A$  和  $B$ , 定义  $B$  对  $A$  的包含度为

$$D_s(A, B) = a^2 \quad (4-3)$$

其中,  $a$  表示满足  $A \subseteq aB$  的最小实数。

显然, 当  $A \subseteq B$  时,  $D_s(A, B) \leq 1$ ; 反之,  $D_s(A, B) > 1$ 。

### 4.2.2 鲁棒预测控制器设计

在系统干扰不随时间逐渐衰减至零的情况下, 系统状态是不可能被稳定在原点的, 控制目标转变为将状态控制在某个原点邻域内。以  $\Omega$  对系统的预测可达集的包含度为优化指标, 采用预测步长逐渐减小的控制策略, 并将  $\Omega$  作为预测控制的终端状态约束集。初始时刻, 预测控制的优化问题  $P_0(x_0)$  可描述为

$$\begin{aligned}
 & \min_{u(i, x_0) \in U} J(u, x_0) \\
 & = \sum_{i=0}^{N-1} D_s(X_i(x_0, u(x_0)), \Omega) + \delta D_s(X_N(x_0, u(x_0)), \Omega) \\
 \text{s.t. } & X_{i+1}(x_0, u(x_0)) = f(X_i(x_0, u(x_0)), u(i, x_0)) + W \\
 & X_i(x_0, u(x_0)) \subseteq X \\
 & X_N(x_0, u(x_0)) \subseteq \Omega
 \end{aligned} \tag{4-4}$$

其中,  $N$  为预测步长,  $\delta > 0$  为一个可调权重系数。定义  $J^*(x_0)$  为  $J(u, x_0)$  的极小值,  $u^*(x_0) = \{u^*(0, x_0), \dots, u^*(N-1, x_0)\}$  为其最优控制输入序列,  $\bar{X} = \{x_0, X_1(x_0, u^*(x_0)), \dots, X_N(x_0, u^*(x_0))\}$  为相应的最优预测可达集序列。在实际控制中, 只将  $u^*(0, x_0)$  作用于实际系统, 下一时刻的输入由下一时刻的优化给出, 如此反复, 得到滚动控制律  $u_{RH} = \{u^*(0, x_0), u^*(0, x_1), \dots\}$ 。

定义  $\Phi = \{x | P_0(x) \text{ 有解}\}$ , 即对  $\forall x \in \Phi$ , 与其对应的预测控制优化问题 [如式 (4-4) 所示] 都有解。 $\Phi$  并不是一个不变集, 即与  $x_1 = f(x_0, u^*(0, x_0)) + w_0$  对应的优化问题  $P_0(x_1)$  不一定有解。这也是我们采用预测步长逐渐减小的控制策略的原因。

$k$  时刻, 有  $x_k = f(x_{k-1}, u^*(0, x_{k-1})) + w_{k-1}$ , 预测控制的优化问题  $P_k(x_k)$  可描述为

当  $k = 1, 2, \dots, N-1$  时,

$$\begin{aligned}
 & \min_{u(i, x_k) \in U} J(u, x_k) \\
 & = \sum_{i=0}^{N-1-k} D_s(X_i(x_k, u(x_k)), \Omega) + \delta D_s(X_{N-k}(x_k, u(x_k)), \Omega) \\
 \text{s.t. } & X_{i+1}(x_k, u(x_k)) = f(X_i(x_k, u(x_k)), u(i, x_k)) + W \\
 & X_i(x_k, u(x_k)) \subseteq X \\
 & X_{N-k}(x_k, u(x_k)) \subseteq \Omega
 \end{aligned} \tag{4-5}$$

当  $k \geq N$  时,

$$\begin{aligned}
 & \min_{u(0, x_k) \in U} J(u, x_k) \\
 & = D_s(x_k, \Omega) + \delta D_s(X_1(x_k, u(0, x_k)), \Omega) \\
 \text{s.t.} \quad & X_1(x_k, u(0, x_k)) = f(x_k, u(0, x_k)) + W \\
 & X_1(x_k, u(0, x_k)) \subseteq \Omega
 \end{aligned} \tag{4-6}$$

$N$  时刻,  $x_N$  已进入  $\Omega$  (如定理 4.1 所示), 因此式 (4-6) 表示的实际上就是  $\Omega$  内的点的优化问题。此时, 预测步长退化到只有一步。

**注 4.1** 在  $\Omega$  内, 控制目标转变为最小化  $D_s(X_1(x_k, u(0, x_k)), \Omega)$ , 而  $D_s(0, \Omega) = 0$ , 意指控制器使得状态的一步预测可达集尽量逼近原点, 这与我们的理想控制目标是一致的。

**定理 4.1** 对  $\forall x_0 \in \Phi$ , 采用滚动控制律  $u_{RH}$ ,  $x_0$  可在  $N$  步内进入  $\Omega$ , 随后一直处于  $\Omega$  中。

证明:

当  $k=1$  时, 预测步长为  $N-1$ ,  $\{u^*(1, x_0), \dots, u^*(N-1, x_0)\}$  是优化问题  $P_1(x_1)$  的一条可行控制序列, 其对应的性能指标  $J(u, x_1)$  是  $P_1(x_1)$  的一个可行解, 因此,  $P_1(x_1)$  有解。

类似地, 当  $k=2, \dots, N-1$  时,  $P_k(x_k)$  都有解。当  $k=N$  时,  $x_N$  进入  $\Omega$ , 又由假设 4.2 可知, 对  $\forall x \in \Omega$ , 都存在  $u \in U$ , 使得  $f(x, u) + W \subseteq \Omega$ 。因此, 当  $k \geq N$  时, 式 (4-6) 表示的优化问题都有解,  $x_k$  也会一直处于  $\Omega$  中。

证毕。

### 4.2.3 预测可达集与终端状态集的计算方法

本小节旨在介绍一种计算预测可达集与终端状态集的简单方法。为后面

章节描述方便, 首先将系统模型式 (4-1) 改写为

$$x_{k+1}^i = f_i(x_k, u_k) + w_k, \quad i=1, 2, \dots, n \quad (4-7)$$

其中,  $x_{k+1}^i$  表示  $x_{k+1}$  的第  $i$  维分量,  $f_i$  表示与之对应的映射关系。

### 1. 预测可达集的计算方法

为方便起见, 假设当前时刻的状态集为  $A$ , 控制量为  $u$ , 那么下一时刻的预测可达集可表示为  $f(A, u) + W$ 。直接求取预测可达集  $f(A, u) + W$  是一件很困难的事情, 需用另一个比较容易求取的近似预测可达集  $\Psi(A, u) + W$  来替代, 而  $\Psi(A, u) + W$  需要满足两个条件。

S1:  $f(A, u) + W \subseteq \Psi(A, u) + W$ 。

S2: 对  $\forall A_1 \subseteq A$ , 都有  $\Psi(A_1, u) + W \subseteq \Psi(A, u) + W$ 。

我们选取简单的多面体作为可达集的描述, 其形式如下:

$$\begin{aligned} \Psi(A, u) + W \\ = \{(x^1, \dots, x^n) \mid x_{\min}^1 \leq x^1 \leq x_{\max}^1, \dots, x_{\min}^n \leq x^n \leq x_{\max}^n\} \end{aligned} \quad (4-8)$$

在式 (4-8) 中,  $x^i, i=1, 2, \dots, n$  表示  $x$  的第  $i$  维分量,  $x_{\min}^i$  表示第  $i$  维分量的下确界,  $x_{\max}^i$  表示第  $i$  维分量的上确界。 $\Psi(A, u) + W$  由  $x_{\min}^i, x_{\max}^i$  确定,  $x_{\min}^i$  和  $x_{\max}^i$  的求取方法如下:

$$x_{\min}^i = \min_{x \in A, w \in W} f_i(x, u) + w \quad (4-9-1)$$

$$x_{\max}^i = \max_{x \in A, w \in W} f_i(x, u) + w \quad (4-9-2)$$

其中,  $i=1, 2, \dots, n$ ,  $f_i$  表示与  $x^i$  对应的映射关系。

显然, 用本小节方法求取的近似预测可达集满足条件 S1 和 S2。



**注 4.2** 用以替代  $f(A, u) + W$  的  $\Psi(A, u) + W$  原则上应该越小越好, 本小节只是介绍一种简单的求取方法, 所得到的结果比文献[45]用区间运算法所得到的近似预测可达集和文献[46]中的对称多胞形都要大一些, 但是本小节的计算量却要小得多。

## 2. 终端状态集的计算方法

与近似预测可达集类似, 终端状态集也可用简单的多面体来近似, 其形式如下:

$$\Omega = \{(x^1, \dots, x^n) \mid x_{\min, \Omega}^i \leq x^i \leq x_{\max, \Omega}^i, \quad i = 1, 2, \dots, n\} \quad (4-10)$$

其中,  $x_{\min, \Omega}^i$  和  $x_{\max, \Omega}^i$  通过求解以下优化问题得到:

$$\begin{aligned} & \max_{x \in A, w \in W} \left( (x_{\max, \Omega}^1 - x_{\min, \Omega}^1) \times \dots \times (x_{\max, \Omega}^n - x_{\min, \Omega}^n) \right) \\ \text{s.t. } & \hat{\Omega} = \{(x^1, \dots, x^n) \mid x_{\min, \Omega}^i \leq x^i \leq x_{\max, \Omega}^i\} \\ & x_{\min, \Omega}^i \leq \min_{x \in \hat{\Omega}, w \in W} \max_{u \in U} f_i(x, u) + w \\ & x_{\max, \Omega}^i \geq \max_{x \in \hat{\Omega}, w \in W} \min_{u \in U} f_i(x, u) + w \\ & x_{\min, \Omega}^i \leq x_{\max, \Omega}^i, \quad i = 1, 2, \dots, n \\ & \hat{\Omega} \subseteq X \\ & 0 \in \hat{\Omega} \end{aligned} \quad (4-11)$$

**注 4.3** 式(4-11)的优化目标是最大化多面体的体积, 第二条和第三条约束表示对  $\Omega$  内的任意点  $x$ , 都存在  $u \in U$ , 使其一步预测可达集  $\Psi(x, u) + W \subseteq \Omega$ , 以保证  $\Omega$  的不变集特性; 最后一条约束表示  $\Omega$  必须包含原点, 因为设计控制器的理想目标是使状态尽量逼近原点。

**注 4.4** 本节中  $\Omega$  的求取方法与近似预测可达集的求取方法是互相对应的, 文献[45]中的终端状态集比本节的要大, 但是对于本节求取的近似预测可达集而言, 文献[45]中终端状态集的“大”是没有意义的, 而且其计算量也大得多。本节只是介绍一种计算近似预测可达集与终端状态集的简单方

法，主要贡献是提出以终端状态约束集对近似预测可达集包含度的预测控制在线优化的优化指标，避免双模控制中控制量的切换问题。

#### 4.2.4 仿真算例

将文献[45,46]中的搅拌釜反应器模型离散化后用作本次仿真模型。

$$C_A(k+1) = C_A(k) + T_s \left( \frac{q}{V} (C_{Af} - C_A(k)) - k_0 \exp\left(-\frac{E}{RT(k)}\right) C_A(k) + w_{C_A} \right)$$

$$T(k+1) = T(k) + T_s \frac{q}{V} (T_f - T(k)) +$$

$$T_s \left( -\frac{\Delta H k_0}{\rho C_p} \exp\left(-\frac{E}{RT(k)}\right) C_A(k) + \frac{UA}{V \rho C_p} (T_c - T(k)) + w_T \right)$$

模型中各参数的具体数值如下：

$$\begin{aligned} \rho &= 1000 \text{ g/l}, \quad C_p = 0.239 \text{ J/gK}, \quad \Delta H = -5 \times 10^4 \text{ J/mol}; \\ E/R &= 8750 \text{ K}, \quad k_0 = 7.2 \times 10^{10} \text{ min}^{-1}, \quad C_{Af} = 1 \text{ mol/l}; \\ q &= 100 \text{ l/min}, \quad T_f = 350 \text{ K}, \quad V = 100 \text{ l}; \\ UA &= 5 \times 10^4 \text{ J/min K} \end{aligned}$$

系统的稳态工作点（也是我们的控制目标）为：

$$C_A^d = 0.5 \text{ mol/l}, \quad T^d = 350 \text{ K}, \quad T_c^d = 300 \text{ K}$$

采样时间取  $T_s = 0.03 \text{ min}$ ，系统的状态约束为  $0.4 \text{ mol/l} \leq C_A \leq 0.6 \text{ mol/l}$ ；输入约束为  $280 \text{ K} \leq T_c \leq 370 \text{ K}$ ；干扰满足界限约束  $|w_{C_A}| \leq 0.1 \text{ mol/(lmin)}$ ， $|w_T| \leq 2 \text{ K/min}$ ；在如此干扰界限下，得到的终端状态集为

$$\Omega = \{(C_A, T) | 0.4 \leq C_A \leq 0.6, 347.47 \leq T \leq 352.8\}$$

取初始预测步长  $N = 12$ ，权重系数  $\delta = 3$ ，在吸引域中选任意点  $x_0 = (0.42, 300)$ ，得到如图 4-1～图 4-3 所示的仿真结果。

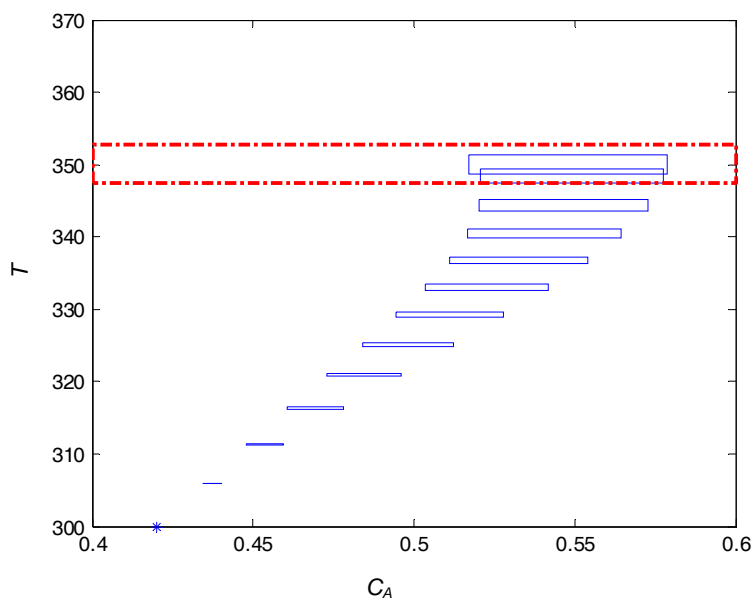


图 4-1 初始点的近似预测可达集序列

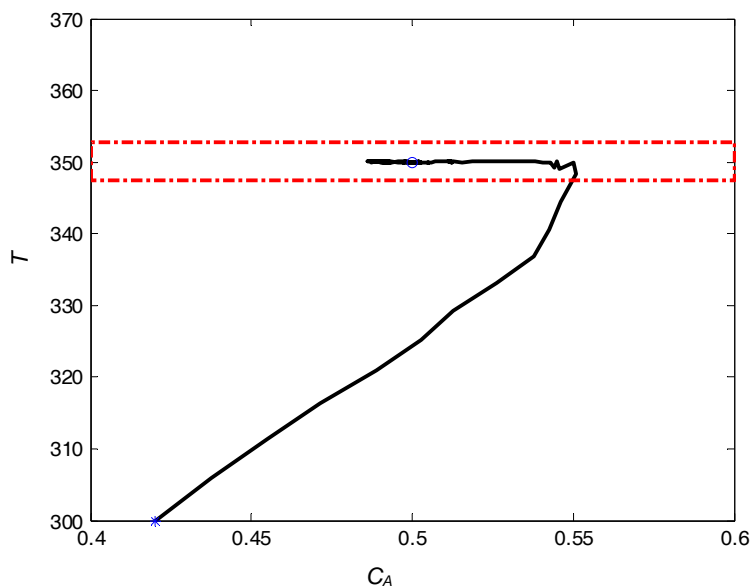


图 4-2 状态收敛曲线图

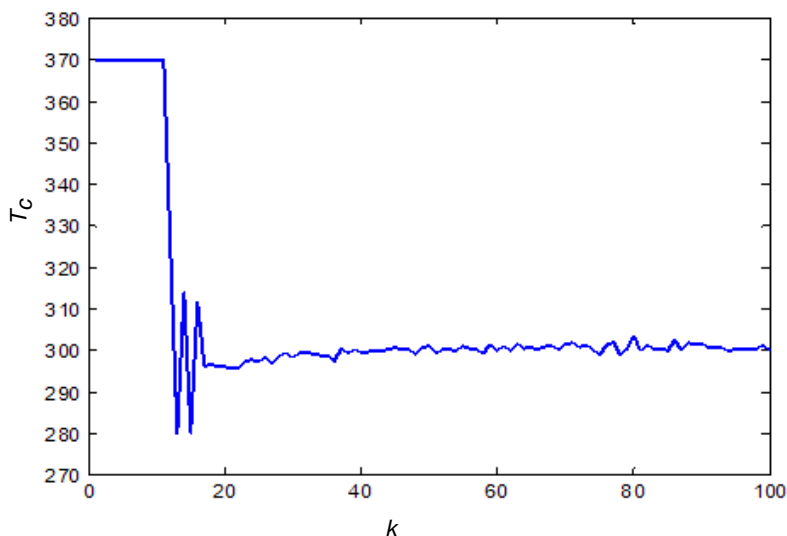


图 4-3 滚动控制作用曲线图

图 4-1 表示  $x_0$  的预测可达集序列，可以看出，末端可达集落在终端状态集（点画线表示的矩形长框）之内。

图 4-2 表示采用预测控制律  $u_{RH}$  时状态的收敛曲线，可以看出，由于系统干扰不随时间衰减至零，状态并未收敛至稳态工作点，只是在其一个小邻域内来回震荡。

图 4-3 给出的是滚动控制作用的变化曲线，很明显，控制量始终满足输入约束。

### 4.3 基于优化线性反馈控制律的鲁棒预测控制

受文献[47]的启发，本节引入输入状态大致稳定的概念，并将预测控制的待优化控制律看成一个线性反馈控制律，再对其线性反馈增益进行在线优化求解，从而设计了一种基于优化线性反馈控制律的 min-max 鲁棒预测控制

器。本节方法相较于文献[48]的方法要更灵活一些，因为文献[48]的方法是先离线计算出一个线性反馈控制增益，将待优化控制律看成一个仿射输入控制（线性反馈控制律与修正量的和），再对修正量进行在线优化求解得到最终控制律。我们从理论上证明了采用本节所设计的鲁棒预测控制器，系统是输入状态大致稳定的，并通过数字仿真验证了其可行性。

### 4.3.1 输入状态大致稳定的概念及相关结论

首先介绍一些相关符号及定义。

函数  $f_1: R^+ \rightarrow R^+$  称为  $\mathcal{K}$  类函数，其中  $R^+ = \{x \in R \mid x \geq 0\}$ ，且其满足连续、严格递增且  $f_1(0) = 0$ 。

函数  $f_2: R^+ \rightarrow R^+$  称为  $\mathcal{K}_\infty$  类函数，当它是  $\mathcal{K}$  类函数且满足  $s \rightarrow +\infty$  时，有  $f_2(s) \rightarrow +\infty$ 。

函数  $f_3: R^+ \times R^+ \rightarrow R^+$  称为  $\mathcal{KL}$  类函数，若其对任意固定的  $d \geq 0$ ，都有  $f_3(\cdot, d)$  为  $\mathcal{K}$  类函数，那么对任意固定的  $s \geq 0$ ，当  $d \rightarrow +\infty$  时，有  $f_3(s, d) \rightarrow +\infty$ 。

**定义 4.1**<sup>[47]</sup> 考虑系统  $x_{k+1} = g(x_k, w_k)$ ，其称为在  $X$  内是输入状态大致稳定的，如果对任意输入  $w_k \in W$ ，任意  $x_0 \in X$ ，都存在一个  $\mathcal{K}_\infty$  类函数  $\alpha(\cdot)$  和一个  $\mathcal{KL}$  类函数  $\beta(\cdot, \cdot)$ ，使得以下不等式成立。

$$\|x_k\|_2 \leq \beta(\|x_0\|_2, k) + \alpha(\bar{W}) \quad (4-12)$$

**引理 4.1**<sup>[47]</sup> 系统  $x_{k+1} = g(x_k, w_k)$  是输入状态大致稳定的，如果存在连续函数  $V: R^n \rightarrow R^+$ ，使得对  $\forall w_k \in W, \forall x_k \in X$ ，都有

$$\alpha_1(\|x_k\|_2) \leq V(x_k) \leq \alpha_2(\|x_k\|_2) + \delta_1(\bar{W}) \quad (4-13)$$

$$V(x_{k+1}) - V(x_k) \leq \delta_2(\bar{W}) - \alpha_3(\|x_k\|_2) \quad (4-14)$$

其中,  $\alpha_1(\cdot), \alpha_2(\cdot), \alpha_3(\cdot)$  为  $\mathcal{K}_\infty$  类函数,  $\delta_1(\cdot), \delta_2(\cdot)$  为  $\mathcal{K}$  类函数;  $V(\cdot)$  称为输入状态大致稳定性李雅普诺夫函数。

引理 4.1 的证明请查阅文献[47]。

### 4.3.2 鲁棒预测控制器设计

对于无干扰系统而言, 为保证系统稳定性, 通常情况下, 在预测控制的优化问题中添加终端代价函数与终端状态集, 使得终端代价函数满足李雅普诺夫函数特性且终端状态集为一个相应的控制不变集<sup>[25]</sup>。对于存在干扰且干扰不随时间衰减至零的系统而言, 为保证系统稳定采取的方法与无干扰系统类似, 只是对终端代价函数的李雅普诺夫函数特性的要求有所放松, 且终端状态集为相应的鲁棒控制不变集。文献[47]针对有界扰动系统给出了一个通用的 min-max 鲁棒预测控制器, 初始时刻对应的优化问题如下:

$$\begin{aligned}
 J_N(\pi, x_0, w) &= \min_{\pi} \max_w \sum_{i=0}^{N-1} q(x(i, x_0), \pi_i(x(i, x_0))) + F(x(N, x_0)) \\
 \text{s.t. } x(i+1, x_0) &= f(x(i, x_0), \pi_i(x(i, x_0))) + w_i \\
 \pi_i(x(i, x_0)) &\in U, x(i, x_0) \in X, \forall w_i \in W, \\
 i &= 0, 1, 2, \dots, N-1, x(N, x_0) \in \Omega
 \end{aligned} \tag{4-15}$$

其中,  $\pi = \{\pi_i(\cdot)\}$ ,  $w = \{w_i\}$ ,  $q(\cdot, \cdot)$  为阶段指标,  $\pi_i(\cdot)$  为待优化控制律,  $F(\cdot)$  为终端代价函数,  $\Omega$  为终端状态集。

**注 4.5** 可以看出, 如果将优化问题式(4-15)中的随机状态  $x(i+1, x_0)$  (因干扰  $w_i$  的存在而导致的状态随机性) 逐一替换成预测可达集, 即将

$$\begin{aligned}
 x(1, x_0) &= f(x(0, x_0), \pi_0(x(0, x_0))) + w_0 \\
 \forall w_0 &\in W
 \end{aligned}$$

$$\begin{aligned}
x(2, x_0) &= f(x(1, x_0), \pi_1(x(1, x_0))) + w_1 \\
&\quad \forall w_1 \in W \\
&\quad \dots \\
x(N, x_0) &= f(x(N-1, x_0), \pi_{N-1}(x(N-1, x_0))) + w_{N-1} \\
&\quad \forall w_{N-1} \in W
\end{aligned}$$

用预测可达集

$$\begin{aligned}
X_1 &= f(x_0, \pi_0(x(0, x_0))) + W \\
X_2 &= f(X_1, \pi_1(X_1)) + W \\
&\quad \dots \\
X_N &= f(X_{N-1}, \pi_{N-1}(X_{N-1})) + W
\end{aligned}$$

替代, 再将阶段指标函数  $q(\cdot, \cdot)$  和终端代价函数  $F(\cdot)$  拓展至集合形式, 则基于 min-max 原理的优化问题式(4-15)就转变为基于预测可达集的优化问题, 如下:

$$\begin{aligned}
&J_N(\pi, x_0, w) \\
&= \min_{\pi} \sum_{i=0}^{N-1} q(X_i, \pi_i(X_i)) + F(X_N) \\
\text{s.t. } &X_{i+1} = f(X_i, \pi_i(X_i)) + W \\
&\pi_i(X_i) \subseteq U, X_i \subseteq X, \quad i = 0, 1, 2, \dots, N-1 \\
&X_N \in \Omega
\end{aligned}$$

因此, 基于 min-max 原理的鲁棒预测控制器与基于预测可达集的鲁棒预测控制器在本质上是—致的。

定义  $J_N^*(x_0)$  为优化问题式(4-15)的最优性能指标, 为保证系统稳定性,  $J_N^*(\cdot)$  需要满足输入状态大致稳定性李雅普诺夫函数特性。因此, 文献[47]对  $\Omega$  的要求不只是满足 RCIS 条件, 其要求为:

对  $\forall x \in \Omega$ ，存在  $u \in U$ ，使得  $\forall w \in W$ ，都有

$$\begin{aligned} f(x, u) + w &\in \Omega \\ F(f(x, u) + w) - F(x) &\leq -q(x, u) + \sigma(\bar{W}^2) \end{aligned}$$

其中， $\sigma(\bar{W}^2)$  为关于  $\bar{W}^2$  的  $\mathcal{K}_\infty$  类函数。

直接求解优化问题式 (4-15) 是很困难的，原因在于当不知道待优化控制律  $\pi_i(\cdot)$  的具体形式时，现有优化算法是无法对优化问题式 (4-15) 进行直接求解的。为了克服这个问题，文献[45,46]直接将  $\pi_i(\cdot)$  看成一个具体控制量，即  $\pi_i(\cdot) = u_i$ （此时， $\pi_i(\cdot)$  不是一个函数，而是一个向量），再对这个控制量进行在线优化求解。文献[48]首先离线计算出一个线性反馈控制律，并将  $\pi_i(\cdot)$  看成一个仿射输入控制律，即  $\pi_i(x) = Kx + e_i$ （ $K$  为离线计算的线性反馈增益， $e_i$  为修正量），再对修正量  $e_i$  进行在线优化求解。可以看出，文献[48]相较于文献[45,46]更优越一些，但是其线性反馈增益是事先离线计算的，而且每一个  $\pi_i(\cdot)$  都用相同的反馈增益，因此，文献[48]的方法还是比较保守的。

受文献[47]的启发，与文献[48]类似，本方法将  $\pi_i(x)$  看成一个线性反馈控制律  $K_i x$ ，再对其线性反馈增益  $K_i$  进行在线优化求解，并取阶段指标函数为  $q(x, u) = x^T Q x$ ，其中， $Q$  为正定矩阵。

初始时刻的优化问题如下：

$$\begin{aligned} J_N(K, x_0, w) &= \min_K \max_w \sum_{i=0}^{N-1} x(i, x_0)^T Q x(i, x_0) + \\ &\quad x(N, x_0)^T P x(N, x_0) \\ \text{s.t. } x(i+1, x_0) &= f(x(i, x_0), K_i x(i, x_0)) + w_i \\ K_i x(i, x_0) &\in U, x(i, x_0) \in X, \forall w_i \in W \\ i &= 0, 1, 2, \dots, N-1, x(N, x_0) \in \Omega \end{aligned} \quad (4-16)$$



其中,  $K = \{K_i\}$ ,  $x(N, x_0)^T P x(N, x_0)$  为终端代价函数。为保证闭环系统稳定, 与文献[47]类似, 我们不只要求  $\Omega$  为鲁棒控制不变集, 还需要满足下述假设 4.3 的条件。

**假设 4.3** 存在某个原点邻域  $\Omega \subseteq X$ 、线性反馈控制律  $u = K_{\text{loc}} x \in U$ 、非负定矩阵  $P$  及一个正实数  $\gamma$ , 使得对  $\forall x \in \Omega, \forall w \in W$ , 都有:  $x_+ \in \Omega$ , 且  $x_+^T P x_+ - x^T P x \leq \gamma \bar{W}^2 - x^T Q x$ 。其中,  $x_+ = f(x, u) + w$ 。

定义  $J_N^*(x_0)$  为优化问题式 (4-16) 的最优性能指标,  $K^* = \{K_i^*\}$  为相应的最优反馈增益。在实际预测控制中, 只有控制量  $u_0 = K_0^* x_0$  真正作用于系统, 而下一时刻的控制量则通过求解其相应时刻的优化问题给出。因此, 滚动控制律可描述为  $u_{\text{RH}}(x) = K_0^* x$ , 其中,  $K_0^*$  为通过在线优化求取的线性反馈增益序列的第一项。

**定理 4.2** 当假设 4.3 成立时, 对  $\forall x_0 \in X$ , 若其对应优化问题式 (4-16) 有解, 则采用滚动控制律  $u_{\text{RH}}(x)$ , 系统是输入状态大致稳定的。

证明分两步: 第一步证明若  $k=0$  时刻, 优化问题式 (4-16) 有解, 则  $k=1$  时刻, 对应的优化问题存在可行解; 第二步证明  $J_N^*(\cdot)$  满足输入状态大致稳定性李雅普诺夫函数特性。

**第一步:**  $k=1$  时刻, 取  $K_f = \{K_1^*, K_2^*, \dots, K_{N-1}^*, K_{\text{loc}}\}$ , 可知

$$x(N-1, x_1) \in \Omega$$

其中

$$x_1 = f(x_0, K_0^* x_0) + w$$

又由假设 4.3 可知, 当  $x(N-1, x_1) \in \Omega$  时, 对  $\forall w \in W$ , 都有

$$x(N, x_1) \in \Omega$$

其中

$$x(N, x_1) = f(x(N-1, x_1), K_{\text{loc}}x(N-1, x_1)) + w$$

因此,  $K_f$  为  $k=1$  时刻的一组可行解。

**第二步:** 将  $K_f$  作用于系统时, 其性能指标  $J_N(K_f, x_1, w)$  满足

$$\begin{aligned} J_N(K_f, x_1, w) - J_N^*(x_0) &= x(N, x_1)^T P x(N, x_1) + x(N-1, x_1)^T Q x(N-1, x_1) - \\ &\quad x(N, x_0)^T P x(N, x_0) - x_0^T Q x_0 \\ &= x(N, x_1)^T P x(N, x_1) + x(N-1, x_1)^T Q x(N-1, x_1) - \\ &\quad x(N-1, x_1)^T P x(N-1, x_1) - x_0^T Q x_0 \end{aligned}$$

已知  $x(N-1, x_1) \in \Omega$ , 由假设 4.3 可知

$$\begin{pmatrix} x(N, x_1)^T P x(N, x_1) + \\ x(N-1, x_1)^T Q x(N-1, x_1) - \\ x(N-1, x_1)^T P x(N-1, x_1) \end{pmatrix} \leq \gamma \bar{W}^2$$

将其代入上式, 有

$$J_N(K_f, x_1, w) - J_N^*(x_0) \leq \gamma \bar{W}^2 - x_0^T Q x_0$$

因此,

$$J_N^*(x_1) - J_N^*(x_0) \leq \gamma \bar{W}^2 - x_0^T Q x_0$$

又知道, 显然存在  $\mathcal{K}_\infty$  类函数  $\alpha_3(\cdot)$ , 其满足

$$\alpha_3(\|x_0\|_2) \leq x_0^T Q x_0$$

所以有

$$J_N^*(x_1) - J_N^*(x_0) \leq \gamma \bar{W}^2 - \alpha_3(\|x_0\|_2)$$

由引理 4.1 可知, 系统是输入状态大致稳定的。

证毕。

### 4.3.3 终端代价函数与终端状态集的求取方法

关于假设 4.3 中终端状态集  $\Omega$ 、非负定矩阵  $P$  的求取方法, 可以查阅文献[48]。 $P$  的求取也可利用  $H_\infty$  控制的思想, 此处只给出大体的计算步骤。

**第一步:** 将系统式 (4-1) 在 0 点处经泰勒一阶展开近似处理成线性系统。

$$x_{k+1} = Ax_k + Bu_k + Cw_k$$

**第二步:** 设计线性反馈控制增益  $K_{\text{loc}}$ , 使得  $A_{\text{KR}} = A + BK_{\text{loc}}$  的特征值都处于单位圆内, 最好使  $A_{\text{KR}}$  的特征值尽量靠近原点。

**第三步:** 求取非负定矩阵  $P$ , 使得  $P$  满足

$$A_{\text{KR}}^T P A_{\text{KR}} - P + Q < 0$$

$$C^T P C - \gamma I < 0$$

另外, 第 2 章中用子集序列逼近最大终端状态集的思想也可用于求取鲁棒预测控制的最大终端状态集, 只是子集序列的构造方法与第 2 章中子集序列的构造方法稍有不同。此处, 我们只给出简单的构造思想, 具体的计算方法不再详述。

定义  $\Omega_{\text{max}}$  为终端代价函数取  $F(x) = x^T P x$  时的最大终端状态集, 基于已经得到的  $\Omega$ , 可以构造出  $\Omega_{\text{max}}$  的另一个比较大的子集  $\Omega_1$ 。

$$\Omega_1 := \{x \in X \mid F(x) \geq F_\Omega^*(x) + x^T Q x - \gamma \bar{W}^2\} \quad (4-17)$$

其中,  $F_\Omega^*(x)$  是下述优化问题的极小值:

$$\begin{aligned}
 & \min_{K_{\text{loc}}} \max_{w \in W} F_{\Omega}(x) \\
 & = q(x, K_{\text{loc}}x) + F(f(x, K_{\text{loc}}x) + w) \\
 & \text{s.t. } f(x, K_{\text{loc}}x) + w \in \Omega
 \end{aligned} \tag{4-18}$$

类似地，另一个子集  $\Omega_2$  可以通过相同的方法构造出来。

$$\Omega_2 := \{x \in X \mid F(x) \geq F_{\Omega_1}^*(x) + x^T Qx - \gamma \bar{W}^2\} \tag{4-19}$$

其中， $F_{\Omega_1}^*(x)$  是下述优化问题的极小值：

$$\begin{aligned}
 & \min_{K_{\text{loc}}} \max_{w \in W} F_{\Omega_1}(x) \\
 & = q(x, K_{\text{loc}}x) + F(f(x, K_{\text{loc}}x) + w) \\
 & \text{s.t. } f(x, K_{\text{loc}}x) + w \in \Omega_1
 \end{aligned} \tag{4-20}$$

如此反复， $\Omega_j, j=3, 4, \dots, \infty$  都可以被构造出来。

$$\Omega_j := \{x \in X \mid F(x) \geq F_{\Omega_{j-1}}^*(x) + x^T Qx - \gamma \bar{W}^2\} \tag{4-21}$$

其中， $F_{\Omega_{j-1}}^*(x)$  是下述优化问题的极小值：

$$\begin{aligned}
 & \min_{K_{\text{loc}}} \max_{w \in W} F_{\Omega_{j-1}}(x) \\
 & = q(x, K_{\text{loc}}x) + F(f(x, K_{\text{loc}}x) + w) \\
 & \text{s.t. } f(x, K_{\text{loc}}x) + w \in \Omega_{j-1}
 \end{aligned} \tag{4-22}$$

与第 2 章中的方法类似，采用支持向量机可从状态空间中依次估计  $\Omega_j, j=1, 2, \dots, \infty$ ，并设置截断函数，中止迭代计算，得出最大终端状态集的最终估计。

**注 4.6** 本节介绍的鲁棒预测控制是基于优化线性反馈控制律的，而对于有些系统来说，其控制输入被约束在幅值很小的范围内，此时可以考虑用饱和控制律（在线性反馈控制律的基础上添加饱和约束）来替代本节的线性反馈控制律。若采用饱和控制律，则需要对预测控制的优化问题进行相应的修改，而且终端状态集也应该转变为相应的饱和控制稳定域。

### 4.3.4 仿真算例

本次仿真采用文献[48]所用模型的离散形式作为仿真模型。

$$\begin{bmatrix} x_1(i+1) \\ x_2(i+1) \end{bmatrix} = \begin{bmatrix} 1 & T \\ T & 1 \end{bmatrix} \begin{bmatrix} x_1(i) \\ x_2(i) \end{bmatrix} + \begin{bmatrix} T\mu \\ T\mu \end{bmatrix} u(i) + \begin{bmatrix} T(1-\mu)x_1(i) \\ -4T(1-\mu)x_2(i) \end{bmatrix} u(i) + 0.1T \begin{bmatrix} 1-x_1(i) \\ 1+4x_2(i) \end{bmatrix} w(i)$$

其中,  $\mu=0.5$ ,  $T=0.1s$ , 状态约束和输入约束分别为  $X=\{x|\|x\|_1 \leq 2\}$ ,  $U=\{u||u| \leq 2\}$ , 干扰  $w$  的 2-范数界为  $\bar{W}=0.2412$ 。

预测控制的预测步长  $N=5$ , 可调参数取  $\gamma=1$ , 阶段指标函数取  $q(x,u)=x^T x$ 。本次仿真中的终端代价函数与终端状态集直接取文献[48]的结果, 即终端代价函数为  $x^T P x$ , 其中

$$P = \begin{bmatrix} 80 & -0.5 \\ -0.5 & 80 \end{bmatrix}$$

终端状态集为

$$\Omega = \{x | x^T P x \leq 4.01\}$$

$\Omega$  内的线性反馈增益为

$$K_{\text{loc}} = [-3.1648 \quad -3.1648]$$

图 4-4 表示吸引域中的任意初始点在采用本节预测控制算法时的收敛曲线。

在图 4-4 中, 细实线围成的椭圆区域表示终端状态集, 较粗的实线表示状态的收敛曲线。由状态的收敛曲线可以看出, 采用本节所设计的鲁棒预测控制器, 吸引域内的任意初始点都可被引导至原点的一个小邻域内。

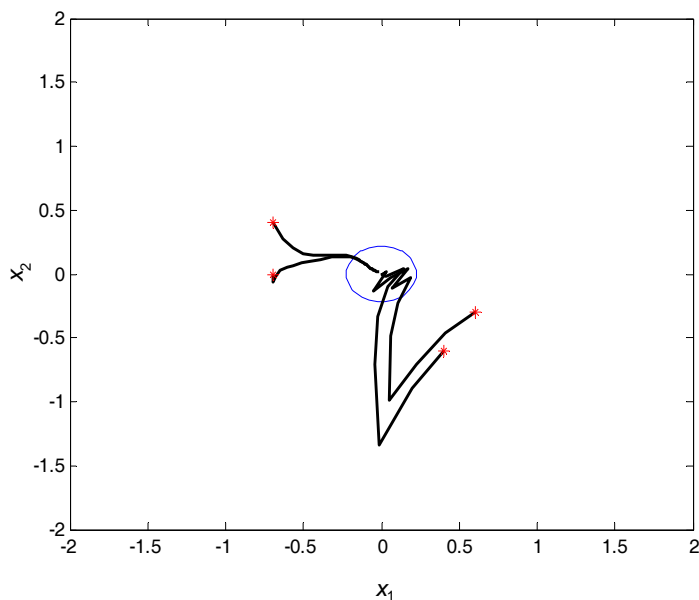


图 4-4 状态收敛曲线

## 4.4 基于优化多项式反馈控制律的鲁棒预测控制

在 4.3 节中介绍了一种基于优化线性反馈控制律的 min-max 鲁棒预测控制算法，其基本思想是将待优化控制律  $\pi_i(x)$  看成一个线性反馈控制律  $K_i x$ ，再对其线性反馈增益  $K_i$  进行在线优化求解。这种方法简单明了、易于实现，但是仅仅将待优化控制律看成一个线性反馈控制律是比较保守的，直接对  $\pi_i(x)$  进行优化求取又是不可能的，因为  $\pi_i(x)$  的形式是未知的。如果  $\pi_i(x)$  连续，则可借助魏尔斯特拉斯第一定理对  $\pi_i(x)$  进行多项式逼近。

**假设 4.4** 每一个  $\pi_i(x)$ ,  $i = 0, 1, 2, \dots, N-1$  都是连续函数。

$\pi_i(x)$  连续且都定义在  $X$  上， $X$  是紧的即意味着其为有界闭集。魏尔斯特拉斯第一定理指出，任意定义在有界闭区间上的连续函数都可以用多项式

来逼近。因此,当假设 4.4 成立时,  $\pi_i(x)$  可由多项式去逼近。由此我们得到启发,采用多项式来逼近  $\pi_i(x)$ ,再对其各项系数进行在线优化。折中考虑逼近精度与计算负担两方面因素,本节用一种结构简单的三次多项式来逼近预测控制的待优化控制律,设计了一种输入状态大致稳定的鲁棒预测控制器,其在线优化的待优化参数为表征控制律的三次多项式的各项系数。为了说明基于预测可达集的预测控制与基于 min-max 原理的预测控制在本质上的一致性,定义了集合与向量之间的距离,以近似预测可达集与原点之间的距离为优化指标,将文献[45,46]中预测可达集的概念引入输入状态大致稳定的鲁棒预测控制中,并采用文献[45]中的扩展区间算法求取预测控制的近似预测可达集。

#### 4.4.1 相关定义

首先定义与 4.2 节类似的预测可达集(在 4.2 节中,预测可达集的定义是基于单个控制量的,而此处预测可达集的定义是基于状态反馈控制的)。考虑集合  $A \subset R^n$ ,  $B \subset R^n$ , 向量  $x \in R^n$ ,  $m \times n$  维矩阵  $K$ , 映射  $g_1(\cdot): R^n \rightarrow R^m$ ,  $g_2(\cdot, \cdot): R^{n+m} \rightarrow R^n$ 。一些关于集合与向量的运算定义如下:

$$x + A = \{x + a \mid a \in A\}, A + B = \{a + b \mid a \in A, b \in B\},$$

$$KA = \{Ka \mid a \in A\}, g_1(A) = \{g_1(a) \mid a \in A\},$$

$$g_2(A, g_1(A)) = \{g_2(a, g_1(a)) \mid a \in A\}$$

**定义 4.2** 考虑集合  $A \subset R^n$ , 向量  $x \in R^n$ ,  $n \times n$  维正定矩阵  $P$ , 称  $D_P(A, x)$  为  $A$  与  $x$  之间的  $P$  距离。同理,对  $n \times n$  维正定矩阵  $Q$ , 称  $D_Q(A, x)$  为  $A$  与  $x$  之间的  $Q$  距离。其中

$$D_P(A, x) = \max \left\{ (a - x)^T P (a - x) \mid a \in A \right\} \quad (4-23)$$

$$D_Q(A, x) = \max \left\{ (a - x)^T Q (a - x) \mid a \in A \right\} \quad (4-24)$$

很明显, 对  $\forall a \in A, \forall B \subseteq A$ , 都有

$$D_P(a, x) \leq D_P(A, x) \quad (4-25)$$

$$D_P(B, x) \leq D_P(A, x) \quad (4-26)$$

**定义 4.3** 考虑预测步长为  $N$ , 已知  $k$  时刻的初始状态为  $x_k$ , 将控制律序列  $\pi = \{\pi_0(x_k), \pi_1(\cdot), \dots, \pi_{N-1}(\cdot)\}$  作用于系统, 系统式 (4-1) 在  $k+i+1$  ( $i \geq 0$ ) 时刻的预测可达集定义为

$$X_{i+1}(x_k, \pi) = f(X_i(x_k, \pi), \pi_i(X_i(x_k, \pi))) + W \quad (4-27)$$

其中

$$X_0(x_k, \pi) = x_k$$

为方便起见, 将式 (4-27) 改写为

$$X_{i+1}(x_k, \pi) = f_{\pi_i}(X_i(x_k, \pi)) + W \quad (4-28)$$

与 4.2 节类似, 直接求取  $X_{i+1}(x_k, \pi)$  所需的计算量是非常大的。为了降低计算负担, 需用其他简单映射  $\tilde{f}_{\pi_i}(\cdot)$  对  $f_{\pi_i}(\cdot)$  进行近似处理, 而这种近似处理需要满足两个条件。

S1——包含性:  $f_{\pi_i}(A) \subseteq \tilde{f}_{\pi_i}(A)$ 。

S2——单调性:  $\forall B \subseteq A, \tilde{f}_{\pi_i}(B) \subseteq \tilde{f}_{\pi_i}(A)$ 。

基于这种近似处理, 预测可达集可由一些比较容易获取的集合来逼近。

**定义 4.4** 考虑预测步长为  $N$ , 已知  $k$  时刻的系统状态为  $x_k$ , 将控制律序列  $\pi = \{\pi_0(x_k), \pi_1(\cdot), \dots, \pi_{N-1}(\cdot)\}$  作用于系统, 系统在  $k+i+1$  ( $i \geq 0$ ) 时刻的近似预测可达集定义为

$$\hat{X}_{i+1}(x_k, \pi) = \tilde{f}_{\pi_i}(\hat{X}_i(x_k, \pi)) + W \quad (4-29)$$



其中

$$\hat{X}_0(x_k, \pi) = x_k$$

由近似预测可达集的定义, 显然可知

$$X_{i+1}(x_k, \pi) \subseteq \hat{X}_{i+1}(x_k, \pi) \quad (4-30)$$

我们采用文献[45]中的扩展区间算法对  $f_{\pi_i}(\cdot)$  进行近似处理。

#### 4.4.2 区间算法简介

顾名思义, 区间算法是定义在区间上的数学运算, 其 4 个基本公式为<sup>[112]</sup>

$$[a, b] + [c, d] = [a + c, b + d] \quad (4-31)$$

$$[a, b] - [c, d] = [a - d, b - c] \quad (4-32)$$

$$[a, b] \times [c, d] = [\min(a \cdot c, a \cdot d, b \cdot c, b \cdot d), \max(a \cdot c, a \cdot d, b \cdot c, b \cdot d)] \quad (4-33)$$

$$[a, b] / [c, d] = [a, b] \times \left[ \frac{1}{d}, \frac{1}{c} \right], 0 \notin [c, d] \quad (4-34)$$

以这 4 个基本公式为基础, 可将区间算法进行扩展, 得到诸如正弦、余弦、指数、对数、开平方等的区间运算结果。此处不对其进行一一详述, 具体算法请查阅文献[114]。区间算法的 4 个基本公式及其扩展至其他映射的结果统称为扩展区间算法。

**引理 4.2**<sup>[113]</sup> 采用扩展区间算法对  $f_{\pi_i}(\cdot)$  进行近似处理, 可满足条件 S1 (包含性) 与 S2 (单调性)。

关于引理 4.2 的证明请查阅文献[113]。由引理 4.2 可知, 使用扩展区间算法得到的集合可作为系统的近似预测可达集。

### 4.4.3 鲁棒预测控制器设计

前面已经提到,当假设 4.4 成立时,待优化控制律  $\pi_i(x)$  可由多项式去逼近。综合考虑计算量与逼近精度两方面因素,我们采用一种结构简单的三次多项式来逼近  $\pi_i(x)$ 。为方便起见,将  $\pi_i(x)$  表示成其向量形式

$$\pi_i(x) = (\pi_i^1(x), \pi_i^2(x), \dots, \pi_i^m(x))^T \quad (4-35)$$

在式 (4-35) 中,  $m$  为控制量的维数,对每一维上的  $\pi_i^j(x)$ ,  $j=1,2,\dots,m$ , 用如下三次多项式逼近:

$$\pi_i^j(x) = \sum_{r=1}^n K_{(i,j),0}^r + K_{(i,j),1}^r x^r + K_{(i,j),2}^r (x^r)^2 + K_{(i,j),3}^r (x^r)^3 \quad (4-36)$$

其中,  $n$  为状态量的维数,  $x^r$  表示  $x$  在第  $r$  维上的分量,  $K_{(i,j),0}^r, K_{(i,j),1}^r, K_{(i,j),2}^r, K_{(i,j),3}^r$  为第  $i$  个预测时刻第  $j$  维控制分量上对应于第  $r$  维状态分量的多项式系数。

将  $\pi_i(x)$  经过三次多项式近似处理后,对  $\pi_i(x)$  的在线优化转变为对三次多项式的系数进行在线优化。可以看出,文献[45,46,48]所采用的方法都可以纳为此方法的一种特殊形式。当预测步长为  $N$  时,此方法的在线优化参数个数为  $N \times n \times m \times 4$ ,而文献[45,46,48]的待优化参数个数为  $N \times m$ 。因此,此方法在增加灵活性的同时,也相应地增加了在线优化的计算负担。

**假设 4.5** 对于正定矩阵  $Q$ , 存在原点邻域  $\Omega \subseteq X$ 、线性反馈控制律  $u = K_{\text{loc}} x \in U$ 、正定矩阵  $P$  及一个正实数  $\gamma$ , 使得对  $\forall A \subseteq \Omega$ , 都有

$$A^+ = \tilde{f}_{K_{\text{loc}}}(A) + W \subseteq \Omega$$

$$D_P(A^+, 0) - D_P(A, 0) \leq \gamma \bar{W}^2 - D_Q(A, 0)$$

其中,  $\tilde{f}_{K_{\text{loc}}}(A)$  表示用于逼近  $f_{K_{\text{loc}}}(A) = f(A, K_{\text{loc}} A)$  的扩展区间算法。

以三次多项式表示的控制律为待优化对象,以近似预测可达集与原点的

距离为优化指标, 基于优化多项式反馈控制律的预测控制器在0时刻对应的具体优化问题可描述为

$$\begin{aligned}
 J_N(\pi, x_0, w) &= \min_{\pi} \sum_{i=0}^{N-1} D_Q(\hat{X}_i(x_0, \pi), 0) + D_P(\hat{X}_N(x_0, \pi), 0) \\
 \text{s.t. } \hat{X}_{i+1}(x_0, \pi) &= \tilde{f}_{\pi_i}(\hat{X}_i(x_0, \pi)) + W \\
 \hat{X}_0(x_0, \pi) &= x_0, \hat{X}_N(x_0, \pi) \subseteq \Omega \\
 \hat{X}_i(x_0, \pi) &\subseteq X, \pi_i(\hat{X}_i(x_0, \pi)) \subseteq U \\
 i &= 0, 1, 2, \dots, N-1
 \end{aligned} \tag{4-37}$$

其中,  $\tilde{f}_{\pi_i}(\hat{X}_i(x_0, \pi))$  表示采用扩展区间算法对  $f_{\pi_i}(\hat{X}_i(x_0, \pi))$  进行近似处理的集合映射;  $f_{\pi_i}(\hat{X}_i(x_0, \pi)) = f(\hat{X}_i(x_0, \pi), \pi_i(\hat{X}_i(x_0, \pi)))$ ;  $\pi_i(\hat{X}_i(x_0, \pi))$  表示状态量空间至控制量空间的集合映射, 其形式如式(4-35)和式(4-36)所示;  $D_Q(\hat{X}_i(x_0, \pi), 0)$  为阶段指标函数,  $D_P(\hat{X}_N(x_0, \pi), 0)$  为终端代价函数,  $\Omega$  为终端状态集。

定义  $J_N^*(x_0, \pi^*, w)$  为优化问题式(4-37)的极小值,  $\pi^* = \{\pi_i^*(\cdot)\}$  为相应的最优控制律,  $\{\hat{X}_i(x_0, \pi^*), i=1, 2, \dots, N\}$  为采用  $\pi^*$  时相应的近似可达集序列。在实际预测控制中, 采用的是滚动控制律  $u_{RH}(x) = \pi_0^*(x)$ 。

**定理 4.3** 对  $\forall x_0 \in X$ , 若其对应的优化问题式(4-37)有解, 则当采用滚动控制律  $u_{RH}(x)$  时, 系统是输入状态大致稳定的。

证明分两步: 第一步证明若  $k=0$  时刻优化问题式(4-37)有解, 则  $k=1$  时刻对应的优化问题存在可行解; 第二步证明  $J_N^*(\cdot)$  为输入状态稳定李雅普诺夫函数。

**第一步: 可行解的存在。**  $k=1$  时刻,  $x_1 = f(x_0, \pi_0^*(x_0)) + w_0$ , 取  $\pi = \{\pi_1^*(\cdot), \pi_2^*(\cdot), \dots, \pi_{N-1}^*(\cdot), \pi_N(\cdot)\}$ , 其中,  $\pi_N(\cdot)$  表示  $\Omega$  内的线性反馈控制律, 即  $\pi_N(x) = K_{loc}x$ 。

定义  $\{\hat{X}_i(x_1, \pi), i=1, 2, \dots, N\}$  为初始状态为  $x_1$  时采用控制律序列  $\pi$  的近似预测可达集序列。因  $x_1 \in \hat{X}_1(x_0, \pi^*)$ , 显然有

$$\hat{X}_i(x_1, \pi) \subseteq \hat{X}_{i+1}(x_0, \pi^*), \quad i=1, 2, \dots, N-1$$

已经知道,  $\hat{X}_N(x_0, \pi^*) \subset \Omega$  且  $\hat{X}_{N-1}(x_1, \pi) \subseteq \hat{X}_N(x_0, \pi^*)$ , 则由假设 4.5 和  $\tilde{f}_{\pi_N}$  的单调性可知

$$\hat{X}_N(x_1, \pi) = \tilde{f}_{\pi_N}(\hat{X}_{N-1}(x_1, \pi)) + W \subset \Omega$$

因此,  $\pi$  为  $k=1$  时刻的一组可行解。

**第二步:**  $J_N^*(\cdot)$  为输入状态稳定李雅普诺夫函数。很明显, 由  $J_N^*(\cdot)$  的定义可知,  $J_N^*(x_0) \geq (\|x_0\|_2)^2$ 。优化问题式 (4-37) 有解表示  $J_N^*(x_0)$  有界, 所以存在正实数  $c_1, c_2$ , 使得  $\forall x_0 \in X_\Omega$ , 都有

$$J_N^*(x_0) \leq c_1 \cdot (\|x_0\|_2)^2 + c_2 \cdot (\bar{W})^2$$

将  $\pi$  作用于系统时, 由于  $\hat{X}_i(x_1, \pi) \subseteq \hat{X}_{i+1}(x_0, \pi^*), i=1, 2, \dots, N-1$ , 由式 (4-25) 和式 (4-26) 可知

$$D_Q(\hat{X}_i(x_1, \pi), 0) \leq D_Q(\hat{X}_{i+1}(x_0, \pi^*), 0)$$

所以性能指标  $J_N(\pi, x_1, w)$  满足

$$\begin{aligned} & J_N(\pi, x_1, w) - J_N^*(x_0, \pi^*, w) \\ & \leq D_P(\hat{X}_N(x_1, \pi), 0) + D_Q(\hat{X}_{N-1}(x_1, \pi), 0) - \\ & \quad D_P(\hat{X}_N(x_0, \pi^*), 0) - D_Q(\hat{X}_0(x_0, \pi^*), 0) \\ & \leq D_P(\hat{X}_N(x_1, \pi), 0) + D_Q(\hat{X}_{N-1}(x_1, \pi), 0) - \\ & \quad D_P(\hat{X}_{N-1}(x_1, \pi), 0) - D_Q(\hat{X}_0(x_0, \pi^*), 0) \end{aligned}$$

已知  $\hat{X}_{N-1}(x_1, \pi) \subseteq \Omega$ , 由假设 4.5 可知

$$\begin{pmatrix} D_P(\hat{X}_N(x_1, \pi), 0) + D_Q(\hat{X}_{N-1}(x_1, \pi), 0) - \\ D_P(\hat{X}_{N-1}(x_1, \pi), 0) \end{pmatrix} \leq \gamma \bar{W}^2$$

将其代入上式, 有

$$\begin{aligned} J_N(\pi, x_1, w) - J_N^*(x_0, \pi^*, w) \\ \leq \gamma \bar{W}^2 - D_Q(\hat{X}_0(x_0, \pi^*), 0) \\ = \gamma \bar{W}^2 - x_0^T Q x_0 \end{aligned}$$

考虑到存在  $\mathcal{K}_\infty$  类函数  $\alpha_3(\cdot)$ , 其满足

$$\alpha_3(\|x_0\|_2) \leq x_0^T Q x_0$$

因此有

$$J_N^*(x_1) - J_N^*(x_0) \leq \gamma \bar{W}^2 - \alpha_3(\|x_0\|_2)$$

由引理 4.1 可知, 系统是输入状态稳定的。

证毕。

**注 4.7** 与注 4.6 介绍的方法类似, 本节的多项式反馈控制律也可以考虑添加饱和约束。同样, 只需对预测控制的优化问题进行相应的修改, 在计算终端状态集时, 也应该将终端状态集考虑为饱和控制的稳定域。

#### 4.4.4 仿真算例

采用 4.3 节中的例子作为本次仿真模型, 各仿真条件不变, 预测步长为  $N=7$ , 可调参数取  $\gamma=1$ , 阶段指标函数取  $q(x, u) = x^T x$ , 终端代价函数仍取  $F(x) = x^T P x$ , 其中

$$P = [80 \quad -0.5; -0.5 \quad 80]$$

终端状态集为

$$\Omega = \{x \mid |x_1| \leq 0.1588, |x_2| \leq 0.1588\}$$

从预测控制的吸引域中取任意初始点，观察采用本节预测控制方法的收敛性。本次仿真初始点为  $x(0) = (0.5, -0.8)^T$ ，图 4-5 表示  $k=0$  时刻的近似预测可达集，图 4-6 表示采用本节鲁棒预测控制方法的状态收敛曲线。

在图 4-5 与图 4-6 中，点画线围起来的矩形区域表示终端状态集  $\Omega$ 。图 4-5 中的细实线小区域为近似预测可达集，图 4-6 中的实线表示状态的收敛曲线。由图 4-5 与图 4-6 可以看出，在  $k=0$  时刻，若末端近似可达集进入  $\Omega$ ，则状态可被引导至原点的某个小邻域内。

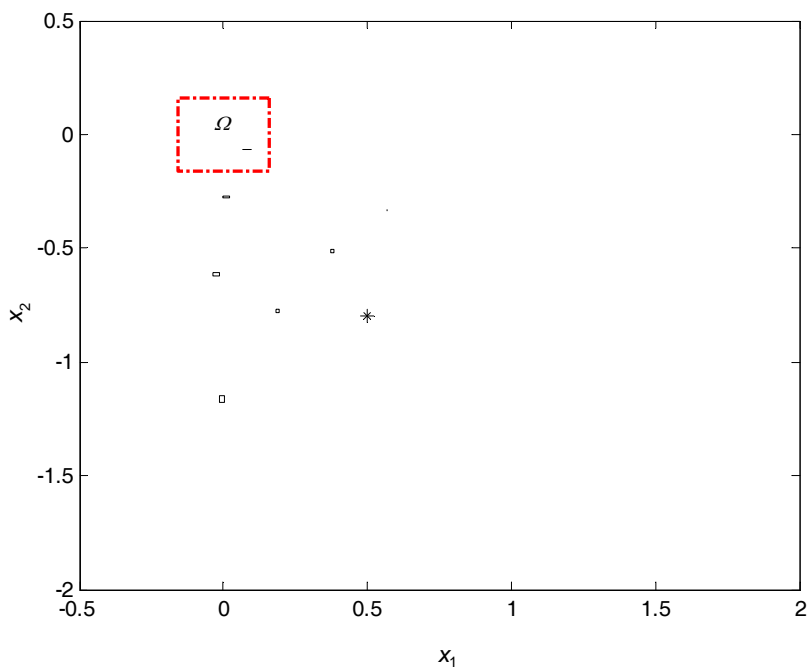


图 4-5 初始时刻的近似预测可达集

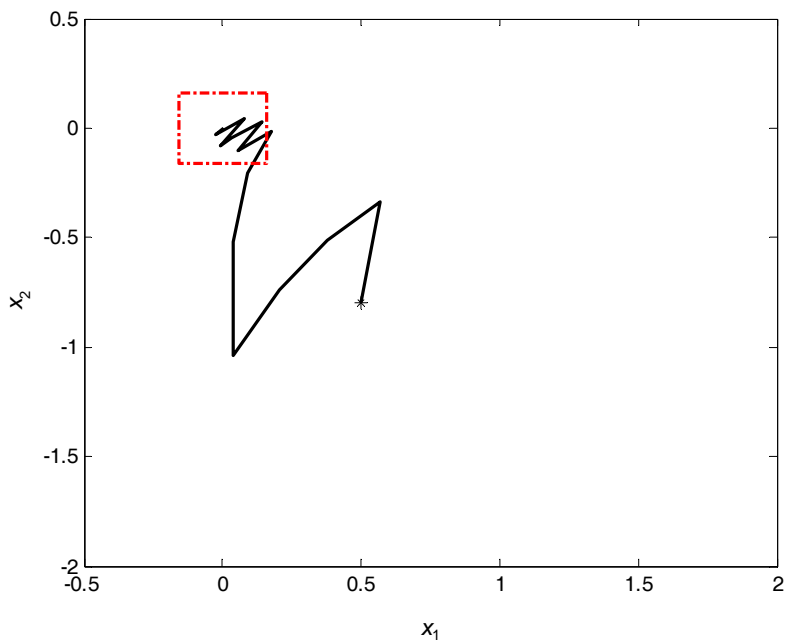


图 4-6 状态收敛曲线

## 4.5 鲁棒预测控制系统中的状态估计研究

前面介绍了存在系统干扰时的几种鲁棒预测控制算法。在这几种鲁棒预测控制算法中,进行在线优化求取的控制律都是基于状态反馈的,也就是说,我们用到了一条假设:状态已知且准确。但是,在实际系统中,这一假设往往不满足,真正用于反馈求取控制律的通常是状态的估计值。因此,在实际系统中,预测控制优化问题中的状态应由其估计值替代,如下(以 min-max 鲁棒预测控制在  $k$  时刻的优化问题为例):

$$\begin{aligned}
& J_N(\pi, \hat{x}_k, w) \\
&= \min_{\pi} \max_w \sum_{i=0}^{N-1} x(i, \hat{x}_k)^T x(i, \hat{x}_k) + x(N, \hat{x}_k)^T P x(N, \hat{x}_k) \\
&\text{s.t. } x(i+1, \hat{x}_k) = f(x(i, \hat{x}_k), \pi_i(x(i, \hat{x}_k))) + w_i \\
&\quad \pi_i(x(i, \hat{x}_k)) \in U, x(i, \hat{x}_k) \in X, \forall w_i \in W, i = 0, 1, 2, \dots, N-1 \\
&\quad x(N, \hat{x}_k) \in \Omega
\end{aligned} \tag{4-38}$$

其中,  $x(0, \hat{x}_k) = \hat{x}_k$ 。可以看出, 如果在  $k$  时刻真实的状态  $x_k$  是未知的, 则可由其估计值  $\hat{x}_k$  替代, 而  $\hat{x}_k$  可以借助滤波器进行状态估计而得到。选取非线性滤波器进行状态估计正是本节的研究内容。

现有用得比较多的非线性滤波器主要是卡尔曼滤波在非线性系统中的一些拓展滤波器, 一般称这类滤波器为卡尔曼滤波体系下的非线性滤波器。但是, 卡尔曼滤波体系下的非线性滤波器包括扩展卡尔曼滤波器 (Extended Kalman Filter, EKF)<sup>[59,60]</sup>、无迹卡尔曼滤波器 (Unscented Kalman Filter, UKF)<sup>[61,62]</sup>、中心差分滤波器 (Central Difference Filter, CDF)<sup>[63-66]</sup>, 都要求系统噪声和量测噪声服从高斯分布。但是, 实际系统中的噪声往往都不服从高斯分布, 甚至有时噪声的统计均值都不为零。为了解决这个问题, 粒子滤波器 (Particle Filter, PF)<sup>[67-71]</sup>应运而生。与卡尔曼滤波体系下的滤波器不同, 基于蒙特卡罗 (Monte Carlo, MC) 方法的粒子滤波器用大量的随机采样点去逼近任意的分布。因此, 粒子滤波器不要求系统噪声和量测噪声服从高斯分布。按照建议分布的不同选取, 现有 4 种典型的粒子滤波器: 直接将预测分布作为建议分布的 PF; 用 EKF 来传递建议分布的扩展卡尔曼粒子滤波器 (Extended Kalman Particle Filter, EKPf)<sup>[72]</sup>; 用 UKF 来传递建议分布的无迹卡尔曼粒子滤波器 (Unscented Kalman Particle Filter, UKPF)<sup>[73]</sup>; 用 CDF 来传递建议分布的中心差分粒子滤波器 (Central Difference Particle Filter, CDPF)<sup>[74]</sup>。

此外, 也有很多学者研究了基于集合理论的集员滤波器 (Set-Membership



Filter, SMF)<sup>[76-81]</sup>及基于模型误差最小准则的预测滤波器(Predictive Filter)<sup>[82-84]</sup>。

本节首先对粒子滤波器进行简单的介绍,然后将 EKPF、UKPF 和 CDPF 的估计精度与计算速度进行综合比较。

### 4.5.1 粒子滤波器简介

粒子滤波器采用蒙特卡罗方法描述已知分布,也就是用大量随机点模拟一个已知分布。在  $k-1$  时刻,假设由状态的后验分布中采样得到  $N$  个粒子点,如果系统不存在过程噪声,那么,当这  $N$  个粒子点通过系统过程传递至  $k$  时刻时,得到的  $N$  个新的粒子点可很好地描述  $k$  时刻的后验分布。但是,事实上,完全准确的系统过程是得不到的,这  $N$  个新的粒子点代表的只是  $k$  时刻后验分布的一个近似,称为预测分布。此时就需要借助  $k$  时刻的量测信息来修正这个预测分布,得到  $k$  时刻真正的后验分布。而贝叶斯理论正是对预测分布进行修正得到后验分布的工具。

#### 1. MC 描述法及贝叶斯理论

前面已经提到,在 PF 中,一个已知分布是用大量随机采样点描述的,即

$$p(x_{0:k} | y_{0:k}) \approx \hat{p}(x_{0:k} | y_{0:k}) = \sum_{i=1}^N \omega_{0,k}^i \delta_{(x_{0,k}^i)}(dx_{0,k}) \quad (4-39)$$

其中,  $\delta(\cdot)$  为狄拉克函数,  $N$  为采样点个数 ( $N$  越大, MC 描述对一个分布的逼近越精确),  $x_{0,k}^i, i=1,2,\dots,N$  为采样点,  $\omega_{0,k}^i$  为采样点的权重系数。

定义  $p(x_k | y_{1:k-1})$  为  $k$  时刻状态的先验分布,  $y_k$  为量测量,则  $k$  时刻状态的后验分布  $p(x_k | y_{1:k})$  可通过贝叶斯理论得到,即

$$p(x_k | y_{1:k}) = \frac{p(y_k | x_k) p(x_k | y_{1:k-1})}{p(y_k | y_{1:k-1})} \quad (4-40)$$

其中

$$p(y_k | y_{1:k-1}) = \int p(y_k | x_k) p(x_k | y_{1:k-1}) dx_k \quad (4-41)$$

为归一化系数。

## 2. 序贯重要性采样

很多时候，我们是无法从状态的后验分布中直接进行采样的，甚至有时后验分布根本就不可知。此时就需要用一个已知且容易采样的分布来逼近后验分布，也就是通常所说的建议分布（也称重要性密度）。假设  $k$  时刻所设计的建议分布为  $q(x_{0:k} | y_{1:k})$ ，我们可以由  $q(x_{0:k} | y_{1:k})$  进行采样得到  $N$  个采样点  $\{x_k^i\}$ ，再利用贝叶斯理论对这  $N$  个采样点进行权重修正，得到加权的  $N$  个采样点。这  $N$  个加权的采样点即可作为对后验分布的一个估计。

$$p(x_{0:k} | y_{1:k}) \approx \sum_{i=1}^N \omega_k^i \delta_{(x_k^i)}(dx_k) \quad (4-42)$$

其中

$$\omega_k^i \propto \frac{p(x_{0:k}^i | y_{1:k})}{q(x_{0:k}^i | y_{1:k})} \quad (4-43)$$

将建议分布  $q(x_{0:k} | y_{1:k})$  写成分解形式

$$q(x_{0:k} | y_{1:k}) = q(x_k | x_{0:k-1}, y_{1:k}) q(x_{0:k-1} | y_{1:k-1})$$

并且考虑到

$$\begin{aligned} p(x_{0:k} | y_{1:k}) &= \frac{p(y_k | x_k) p(x_{0:k} | y_{1:k-1})}{p(y_k | y_{1:k-1})} \\ &= \frac{p(y_k | x_k) p(x_k | x_{k-1})}{p(y_k | y_{1:k-1})} p(x_{0:k-1} | y_{1:k-1}) \\ &\propto p(y_k | x_k) p(x_k | x_{k-1}) p(x_{0:k-1} | y_{1:k-1}) \end{aligned} \quad (4-44)$$

可以得到权重的递推更新公式为

$$\begin{aligned}\omega_k^i &\propto \frac{p(y_k | x_k^i) p(x_k^i | x_{k-1}^i) p(x_{0:k-1}^i | y_{1:k-1})}{q(x_k^i | x_{0:k-1}^i, y_{1:k}) q(x_{0:k-1}^i | y_{1:k-1})} \\ &= \omega_{k-1}^i \frac{p(y_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{0:k-1}^i, y_{1:k})}\end{aligned}\quad (4-45)$$

通过权重的递推更新公式，可以很方便地计算每一个时刻粒子点的权重，从而得到每一个时刻状态的后验分布估计，这种方法称为序贯重要性采样。

### 3. 粒子退化

从权重的递推更新公式中可以看出，随着时间的推移，只有少数一些粒子点甚至只有一个粒子点的权重比较大，而大多数粒子点的权重几乎退化到零，量测噪声很小的时候尤其严重，这种现象称为粒子退化。粒子退化的出现违背了粒子滤波的初衷，大量的计算负担用在了更新那些权重几乎退化到零的粒子上，而这些粒子对我们进行状态估计几乎没有什么贡献。为了克服这个问题，需要对粒子进行重采样。重采样的思想就是剔除小权重的粒子，复制大权重的粒子。假设  $k$  时刻的后验分布由权重为  $\{\omega_k^i\}$  的  $N$  个粒子描述， $\{\omega_k^i\}$  中各个权重值有大有小。经过重采样，这  $N$  个权重为  $\{\omega_k^i\}$  的粒子被  $N$  个权重都为  $1/N$  的新粒子取代。因此，重采样的应用消除了粒子退化的现象，但是这又带来一个新的问题，即这  $N$  个新粒子很多都是相同的，甚至有时它们都是同一个粒子的复制，过程噪声很小的时候尤其严重，这种现象称为粒子贫乏。一种有效的解决方法是采用马尔科夫链蒙特卡罗（Markov Chain Monte Carlo, MCMC）方法，如吉布斯采样器和 Metropolis Hastings 算法，具体算法这里不再给出，请查阅文献[73]。

### 4. 建议分布

另外，一个好的建议分布也可以减轻粒子退化的程度，建议分布越逼近

后验分布，粒子退化的现象就越轻。我们总是希望建议分布能最大限度地逼近后验分布，使得从建议分布中进行采样得到的粒子在描述后验分布时，每个粒子的权重都相差不多。

已知  $x_{0:k-1}$  和  $y_{1:k}$ ，最优的建议分布  $q(x_{0:k} | y_{1:k})$  为

$$q(x_{0:k} | y_{1:k}) = q(x_k | x_{0:k-1}, y_{1:k}) p(x_{0:k-1} | y_{1:k-1}) \quad (4-46)$$

其中

$$q(x_k | x_{0:k-1}, y_{1:k}) = p(x_k | x_{0:k-1}, y_{1:k}) \quad (4-47)$$

由于  $p(x_k | x_{0:k-1}, y_{1:k})$  很难得到，至今，最流行的选择为

$$q(x_k | x_{0:k-1}, y_{1:k}) = p(x_k | x_{k-1})$$

其中， $p(x_k | x_{k-1})$  表示过渡先验分布。

$p(x_k | x_{k-1})$  的求取很简单，但是它并没有用到最新的量测信息  $y_k$ ，这会导致比较严重的粒子退化现象。因此，在设计建议分布时考虑最新的量测信息  $y_k$  吸引了很多学者的目光。其中，文献[72]采用 EKF 来传递粒子以得到建议分布，文献[73]采用 UKF 来传递粒子以得到建议分布，文献[74]采用 CDF 来传递粒子以得到建议分布。下面以用 CDF 来传递粒子为例，介绍如何得到建议分布。

在  $k-1$  时刻，假设从状态的后验分布中进行采样得到权重都为  $1/N$  的  $N$  个粒子  $\{x_{k-1}^i, i=1, 2, \dots, N\}$ 。用 CDF 将这  $N$  个粒子传递至  $k$  时刻，可以得到与此  $N$  个粒子对应的  $N$  个高斯分布估计  $\mathcal{N}(\hat{x}_k^i, \hat{P}_k^i)$ ，其中， $\hat{x}_k^i$  表示这个高斯分布的均值， $\hat{P}_k^i$  表示方差。从每一个高斯分布中进行采样可以得到一个随机粒子  $\tilde{x}_k^i$ ，对  $N$  个高斯分布进行采样可以得到  $N$  个随机粒子  $\{\tilde{x}_k^i\}$ 。对每一个  $\tilde{x}_k^i$  进行权重计算，得到其相应权重  $\tilde{\omega}_k^i$ 。最后，这  $N$  个权重系数为  $\{\tilde{\omega}_k^i\}$  的粒子  $\{\tilde{x}_k^i\}$  即可用来描述  $k$  时刻的建议分布。

$$q(x_k | x_{0:k-1}, y_{1:k}) = \sum_{i=1}^N \tilde{\omega}_k^i \delta_{(\tilde{x}_k^i)}(d\tilde{x}_k) \quad (4-48)$$

对每一个  $\tilde{x}_k^i$  来说, 其权重可以通过下式求取:

$$\tilde{\omega}_k^i \propto \frac{p(\tilde{x}_k^i | x_{k-1}^i)}{q(\tilde{x}_k^i | x_{0:k-1}^i, y_{1:k})} \quad (4-49)$$

其中

$$q(x_k^i | x_{0:k-1}^i, y_{1:k}) = \mathbb{N}(\hat{x}_k^i, \hat{P}_k^i) \quad (4-50)$$

而  $p(x_k^i | x_{k-1}^i)$  则由系统的过程噪声决定。

再依据贝叶斯理论对每一个粒子  $\tilde{x}_k^i$  的权重进行修正, 得到其更新后的权重  $\omega_k^i$ 。

$$\omega_k^i \propto p(y_k | \tilde{x}_k^i) \tilde{\omega}_k^i \quad (4-51)$$

进行权重更新后, 这  $N$  个权重系数为  $\{\omega_k^i\}$  的粒子  $\{\tilde{x}_k^i\}$  即可用来描述  $k$  时刻的状态后验分布。

$$p(x_k | y_{1:k}) \approx \hat{p}(x_k | y_{1:k}) = \sum_{i=1}^N \omega_k^i \delta_{(\tilde{x}_k^i)}(d\tilde{x}_k) \quad (4-52)$$

采用重采样和马尔科夫链蒙特卡罗方法, 最终可以得到表征  $k$  时刻状态后验分布的  $N$  个权重都为  $1/N$  的新粒子  $\{x_k^i\}$ 。

综上所述, 我们以中心差分粒子滤波器为例, 给出其具体的计算步骤<sup>[74]</sup>。

**Step 1:** 设置粒子滤波中的粒子数为  $N$ , 并初始化  $k=0$ 。

对  $i=1:N$ , 从状态的后验分布  $p(x_k)$  中进行采样, 得到  $x_k^i$ 。

**Step 2:** 用 CDF 传递粒子。

对  $i=1:N$ , 运行 CDF 算法<sup>[64,65]</sup>得到  $\hat{x}_{k+1}^i$  和  $\hat{P}_{xx}^i(k+1)$ 。

**Step 3:** 重要性采样。

对  $i=1:N$ ，由  $q(x_{k+1}^i | x_{0:k}^i, y_{1:k+1}) = \mathbb{N}(\hat{x}_{k+1}^i, \hat{P}_{xx}^i(k+1))$  进行采样得到  $\tilde{x}_k^i$ ，计算其权重并进行归一化处理。

$$\omega_{k+1}^i = \frac{p(y_{k+1} | \tilde{x}_{k+1}^i) p(\tilde{x}_{k+1}^i | x_k^i)}{q(\tilde{x}_{k+1}^i | x_{0:k}^i, y_{1:k+1})} \quad (4-53)$$

$$\omega_{k+1}^i = \omega_{k+1}^i \left( \sum_{i=1}^N \omega_{k+1}^i \right)^{-1} \quad (4-54)$$

**Step 4:** 重要性采样。

剔除小权重的粒子，复制大权重的粒子。

**Step 5:** 运行 MCMC 算法<sup>[73]</sup>。

运行 MCMC 算法，得到表征  $k+1$  时刻状态后验分布的  $N$  个权重都为  $1/N$  的新粒子  $\{x_{k+1}^i\}$ 。

**Step 6:** 置  $k=k+1$ ，返回 Step 2。

## 4.5.2 滤波器性能对比仿真

本次仿真采用文献[73,74]中的模型为仿真对象，比较 EKPF、UKPF、CDPF 的估计精度与计算速度。过程方程为

$$x_{k+1} = 1 + \sin(\omega\pi k) + \phi_1 x_k + w_k$$

量测方程为

$$y_k = \begin{cases} \phi_2 x_k^2 + n_k, & k \leq 30 \\ \phi_3 x_k - 2 + n_k, & k > 30 \end{cases}$$

其中，在过程方程中，参数设置为  $\omega = 4 \times 10^{-2}$ ， $\phi_1 = 0.5$ ，过程噪声  $w_k$  为伽马

噪声  $G_a(3,2)$ ；在量测方程中，参数设置为  $\phi_2 = 0.2$ ， $\phi_3 = 0.5$ ，量测噪声  $n_k$  为高斯噪声  $\mathcal{N}(0,0.01)$ 。

取粒子数  $N = 200$ ，区间长度  $h = \sqrt{3}$ ，并设置初始的真实状态为  $x_1 = 1$ ，其估计值取为  $\hat{x}_1 = 0$ 。程序运行 100 次，通过统计计算求取均值，分别得到采用三种滤波器进行状态估计的结果和消耗的计算时间如图 4-7、表 4-1 和表 4-2 所示。

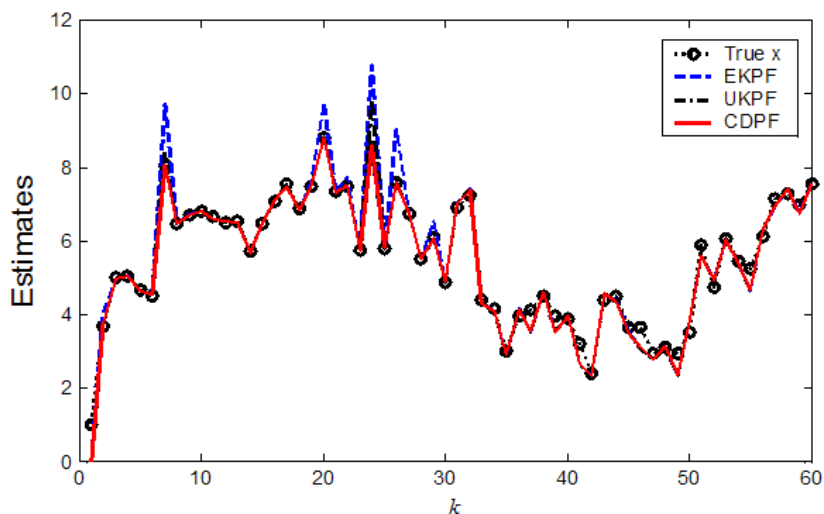


图 4-7 三种滤波器的状态估计比较示意图

表 4-1 三种滤波器的状态估计均方差

Mean MSE of EKPF	Mean MSE of UKPF	Mean MSE of CDPF
0.4973	0.2996	0.2392

表 4-2 三种滤波器消耗的计算时间

Mean time cost of EKPF	Mean time cost of UKPF	Mean time cost of CDPF
6.1603s	19.2578s	13.4645s

从图 4-7、表 4-1 和表 4-2 中可以看出,CDPF 在估计精度上相较于 EKPF 更优越一些,与 UKPF 相当,虽然其消耗的计算时间仍然比 EKPF 多,但相较于 UKPF 却明显减少了。

### 4.5.3 CDPF 在鲁棒预测控制中的应用仿真

鉴于 CDPF 相对于 EKPF 在估计精度上的优势及相对于 UKPF 在计算速度上的优势,以 4.2 节介绍的鲁棒预测控制方法为例,将 CDPF 应用于鲁棒预测控制中的状态估计。仿真模型取 4.2 节中的例子,其中,过程方程为

$$C_A(k+1) = C_A(k) + T_s \left( \frac{q}{V} (C_{Af} - C_A(k)) - k_0 \exp\left(-\frac{E}{RT(k)}\right) C_A(k) + w_{C_A} \right)$$

$$T(k+1) = T(k) + T_s \frac{q}{V} (T_f - T(k)) +$$

$$T_s \left( -\frac{\Delta H k_0}{\rho C_p} \exp\left(-\frac{E}{RT(k)}\right) C_A(k) + \frac{UA}{V \rho C_p} (T_c - T(k)) + w_T \right)$$

过程方程中各参数与 4.2 节中的例子一致。量测方程取

$$y(k+1) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} C_A(k+1) \\ T(k+1) \end{bmatrix} + \begin{bmatrix} v_C \\ v_T \end{bmatrix}$$

量测噪声取  $v_C = \text{rand}(0, 0.03)$ ,  $v_T = \text{rand}(-1, 0.5)$ , 其中,  $\text{rand}(0, 0.03)$  表示区间  $(0, 0.03)$  上的平均分布随机噪声,  $\text{rand}(-1, 0.5)$  表示区间  $(-1, 0.5)$  上的平均分布随机噪声。

与 4.2 节类似,采样时间取  $T_s = 0.03 \text{ min}$ , 将 CDPF 应用于鲁棒预测控制中的状态估计后,得到如图 4-8 所示的仿真结果。



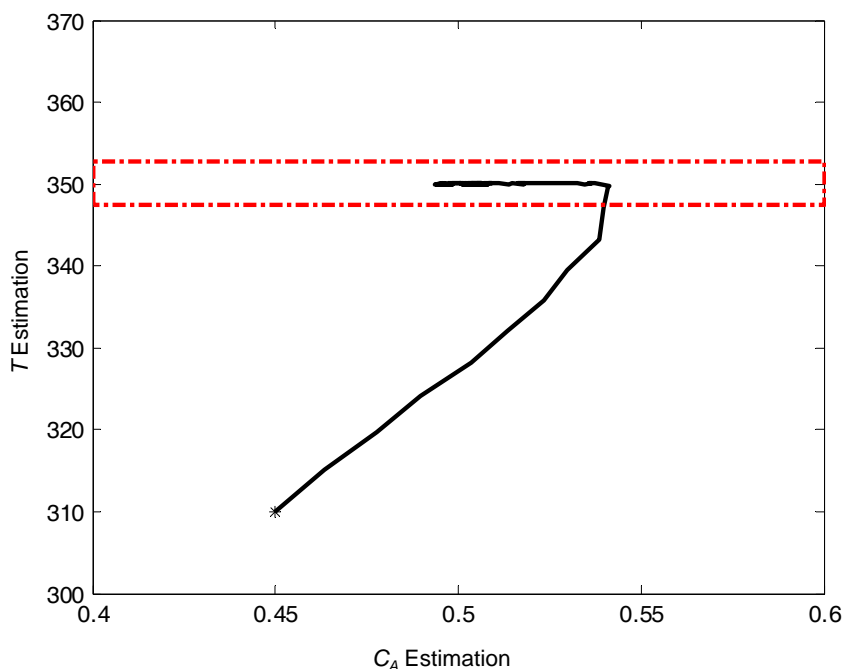


图 4-8 状态收敛曲线图

在图 4-8 中，点画线表示鲁棒预测控制的终端状态集。可以看出，在鲁棒预测控制的优化问题中，将真实的状态用滤波估计状态代替后，系统的估计状态仍可以被控制至原点的某个小邻域内。

## 4.6 本章小结

本章介绍了三种基于扰动模型的鲁棒预测控制算法：以集合包含度为优化指标的鲁棒预测控制；基于优化线性反馈控制律的鲁棒预测控制；基于优化多项式反馈控制律的鲁棒预测控制。在理论上证明了本章所介绍的鲁棒预测控制算法的稳定性，并通过仿真验证了算法的可行性。另外，本章针对非线性系统的状态估计，大体介绍了现有非线性滤波方法的适用范围并对粒子

滤波器进行了简单介绍，比较了几种典型粒子滤波器的估计精度与计算速度，并将 CDPF 滤波器应用于非线性系统鲁棒预测控制的状态估计中。但是，在我们所设计的预测控制系统中，用状态估计值替代真实的状态时，势必会影响系统的稳定性。因时间所限，我们并没有分析滤波的收敛过程对预测控制系统的稳定性会产生何种影响，这一部分内容将是我们下一阶段的研究方向。较之于现有的鲁棒预测控制方法，本章所设计的基于优化线性反馈控制律的鲁棒预测控制方法和基于优化多项式反馈控制律的鲁棒预测控制方法在待优化控制律的选取上更灵活，其鲁棒预测控制系统的稳定域也要大一些。

## 第 5 章

# 跟踪问题中的预测控制

### 5.1 引言

第 2~4 章讨论了针对镇定问题的预测控制，即控制目标为将系统状态引导至平衡点。对很多系统来说，不仅需要研究镇定问题的预测控制，还需要研究跟踪问题的预测控制。文献[85-90]分别针对跟踪问题中的线性系统设计了预测控制器。对于非线性系统在跟踪问题中的预测控制器设计，现有文献并不多见。难点就在于跟踪问题中的非线性预测控制器的稳定性分析相对镇定问题中的稳定性分析来说要难得多。

本章将非线性系统的跟踪问题按期望输出的形式分为两种：对单个稳定期望输出点的逼近问题（也称定值调节问题）；对某条期望输出轨迹的跟踪问题。针对这两种跟踪问题，分别设计了非线性预测控制器，最后针对第二种跟踪问题给出了一种简单的计算预测控制次优解的方法。

## 5.2 对单个稳定期望输出点的逼近

### 5.2.1 问题描述

这种形式的跟踪问题一般指的是要求系统的输出穿越一些事先设置好的稳定期望输出点。例如，要求飞行器由  $A$  点出发，依次经过  $B$  点、 $C$  点、 $D$  点，最后返回  $A$  点。这种形式的跟踪问题最终都可以归纳为对单个期望输出点的逼近，下面以飞行器例子来说明这一问题。初始时刻，可以将控制目标设置为飞行器对  $B$  点的逼近，待飞行器到达  $B$  点时，切换控制目标至对  $C$  点的逼近。因此，本节要解决的问题可陈述为：

对于给定的稳定期望输出  $y_d$ ，设计预测控制器，使得系统输出能最终到达期望输出  $y_d$ 。

考虑非线性系统如下：

$$x_{k+1} = f(x_k, u_k) \quad (5-1)$$

$$y_{k+1} = I_{n \times n} x_{k+1} \quad (5-2)$$

在过程方程中， $x_k \in R^n$ ， $u_k \in R^m$  分别为系统在采样时刻  $k$  的状态和输入； $f(\cdot, \cdot)$  为关于  $x_k, u_k$  的已知非线性连续函数，满足利普希茨有界。系统的状态约束和输入约束分别为  $x_k \in X$ ， $u_k \in U$ ，满足  $X$  和  $U$  都是紧的。在输出方程中， $y_{k+1} \in R^n$  表示系统在采样时刻  $k+1$  时的输出。

为简单起见，将系统式 (5-1)、式 (5-2) 用如下输入输出模型来表示：

$$y_{k+1} = f(y_k, u_k) \quad (5-3)$$

初始时刻, 预测控制的优化问题为

$$\begin{aligned}
 \min_{u(i, y_0) \in U} J(u, y_0) &= \sum_{i=0}^{N-1} q((y(i, y_0) - y_d), u(i, y_0)) + F(y(N, y_0) - y_d) \\
 \text{s.t. } y(i+1, y_0) &= f(y(i, y_0), u(i, y_0)) \\
 y(i+1, y_0) &\in Y, u(i, y_0) \in U, i=0, \dots, N-1 \\
 y(N, y_0) &\in Y_f(y_d)
 \end{aligned} \tag{5-4}$$

其中,  $N$  为预测步长,  $y(0, y_0) = y_0$ ,  $y_d$  表示期望输出,  $Y$  为输出约束集,  $U$  为控制约束集,  $Y_f(y_d)$  为输出的终端状态约束集, 它与期望输出  $y_d$  有关。一般情况下,  $y_d = f(y_d, 0)$  是不能满足的, 所以, 优化问题式 (5-4) 中的阶段指标函数不考虑对控制能量的限制, 形式为  $q(y - y_d, u) = (y - y_d)^T Q(y - y_d)$ ,  $Q$  为对应维数的正定矩阵。此时, 阶段指标函数可以用更简洁的形式描述为  $q(y - y_d) = (y - y_d)^T Q(y - y_d)$ 。

### 5.2.2 稳定性分析

定义跟踪误差  $e$  为系统输出与期望输出的差值, 即

$$e = y - y_d \tag{5-5}$$

将跟踪误差代入系统的输入输出模型式 (5-3), 可得跟踪误差的动态过程为

$$e_{k+1} + y_d = f(e_k + y_d, u_k) \tag{5-6}$$

定义

$$f_e(e_k, u_k) = f(e_k + y_d, u_k) - y_d \tag{5-7}$$

则动态过程式 (5-6) 可以转化为

$$e_{k+1} = f_e(e_k, u_k) \tag{5-8}$$

由于  $f(\cdot, \cdot)$  为已知非线性函数, 而  $y_d$  为已知的期望输出, 因此,  $f_e(\cdot, \cdot)$  是已知的。

引入跟踪误差变量后, 控制目标转变为将跟踪误差控制到零, 此时, 优化问题式 (5-4) 变形为

$$\begin{aligned} \min_{u(i, e_0) \in U} J(u, e_0) &= \sum_{i=0}^{N-1} q(e(i, e_0)) + F(e(N, e_0)) \\ \text{s.t. } e(i+1, e_0) &= f_e(e(i, e_0), u(i, e_0)) \\ e(i+1, e_0) + y_d &\in Y, \quad u(i, e_0) \in U, \quad i=0, \dots, N-1 \\ e(N, e_0) &\in E_f \end{aligned} \quad (5-9)$$

其中,  $E_f$  为跟踪误差的终端状态集。

定义  $J(u, e_0)$  的极小值为  $J^*(e_0)$ , 最优输入序列为

$$u^*(e_0) = \{u^*(0, e_0), \dots, u^*(N-1, e_0)\}$$

最优跟踪误差序列为

$$e^*(e_0) = \{e^*(1, e_0), \dots, e^*(N, e_0)\}$$

在实际控制中, 只将  $u^*(0, x_0)$  作用于实际系统, 下一时刻的输入由下一时刻的优化给出, 如此反复, 得到滚动控制律

$$u_{RH} = \{u^*(0, e_0), u^*(0, e_1), \dots\}$$

其中

$$e_1 = f_e(e_0, u^*(0, e_0))$$

考虑如下假设。

**假设 5.1** 存在原点 (此处的原点意指  $e=0$ ) 的一个邻域  $E_f$ , 满足:

(TC1)  $E_f$  为包含零点的闭集, 并且对任意的  $e \in E_f$ , 都有  $e + y_d \in Y$ 。

(TC2) 对任意的  $e \in E_f$  , 都存在  $u_{\text{loc}} \in U$  , 使得  $F(e) \geq q(e) + F(f_e(e, u_{\text{loc}}))$  成立, 并且  $f_e(e, u_{\text{loc}}) \in E_f$ 。

基于假设 5.1, 可以得到以下结论。

**定理 5.1** 当假设 5.1 成立时, 如果初始跟踪误差  $e_0$  满足  $e_0 + y_d \in Y$  且  $e^*(N, e_0) \in E_f$  , 则采用滚动控制律  $u_{\text{RH}}$  可使  $e_0$  最终收敛到零。

证明:

已经知道  $k=0$  时刻的最优输入序列为

$$u^*(e_0) = \{u^*(0, e_0), \dots, u^*(N-1, e_0)\}$$

最优跟踪误差序列为

$$e^*(e_0) = \{e^*(1, e_0), \dots, e^*(N, e_0)\}$$

将  $u^*(0, e_0)$  作用于系统, 过渡到下一时刻, 跟踪误差为  $e_1 = f_e(e_0, u^*(0, e_0))$ , 构造输入序列

$$u(e_1) = \{u^*(1, e_0), u^*(2, e_0), \dots, u^*(N-1, e_0), u_{\text{loc}}\}$$

其中,  $u_{\text{loc}}$  为满足

$$F(e^*(N, e_0)) \geq q(e^*(N, e_0)) + F(f_e(e^*(N, e_0), u_{\text{loc}}))$$

并且

$$f_e(e^*(N, e_0), u_{\text{loc}}) \in E_f$$

的控制输入。

很明显,  $u(e_1)$  是系统在  $k=1$  时刻的一组可行控制律。而与  $u(e_1)$  对应的性能指标  $J(u(e_1), e_1)$  满足

$$J(\mathbf{u}(e_1), e_1) - J^*(e_0) = q(e^*(N, e_0)) + F(f_e(e^*(N, e_0), u_{\text{loc}})) - q(e_0) - F(e^*(N, e_0))$$

因为

$$F(e^*(N, e_0)) \geq q(e^*(N, e_0)) + F(f_e(e^*(N, e_0), u_{\text{loc}}))$$

所以有

$$\begin{aligned} J^*(e_1) - J^*(e_0) &\leq J(\mathbf{u}(e_1), e_1) - J^*(e_0) \\ &\leq -q(e_0) \\ &< 0 \end{aligned}$$

因此，当采用滚动控制律  $\mathbf{u}_{\text{RH}}$  时， $e_0$  可最终收敛到零。

证毕。

### 5.2.3 终端代价函数的求取

$y_d$  是稳定的期望输出点，意味着存在这样一个（或几个） $\bar{u} \in U$ ，使得  $0 = f_e(0, \bar{u})$ ，即  $y_d = f(y_d, \bar{u})$  成立。

考虑如下假设。

**假设 5.2** 使得  $0 = f_e(0, \bar{u})$  成立的  $\bar{u}$  是可求的。

当假设 5.2 成立时，可先求出  $\bar{u}$ ，再定义  $\Delta u_k = u_k - \bar{u}$ ，将跟踪误差的动态过程式 (5-8) 转换为

$$e_{k+1} = f_e(e_k, \Delta u_k + \bar{u}) = f_{e\bar{u}}(e_k, \Delta u_k) \quad (5-10)$$

此时，可采用与第 3 章中求取终端代价函数相同的方法来求取跟踪误差的终端代价函数。将式 (5-10) 沿 (0,0) 泰勒展开，进行线性化处理，得到其近似线性模型如下：



$$e_{k+1} = A e_k + B \Delta u_k \quad (5-11)$$

其中

$$A = (\partial f_{e\bar{u}} / \partial e)(0,0)$$

$$B = (\partial f_{e\bar{u}} / \partial u)(0,0)$$

基于近似线性模型式 (5-11)，可利用线性最优控制理论求得一个终端代价函数，具体计算方法见第 3 章，此处不再详述。

另外，与第 3 章中的定理 3.1 类似，在跟踪问题中，按比例增大终端代价函数，跟踪误差的终端状态约束集也会增大。

## 5.2.4 终端状态约束集的求取

首先依据条件 (TC1) 确定跟踪误差的约束范围，记为  $E$ ， $E$  需要满足：对任意的  $e \in E$ ，都有  $e + y_d \in Y$ 。得到  $E$  后，可以借鉴第 2 章中求取终端状态约束集的方法来求取跟踪问题下的跟踪误差的终端状态约束集。

定义  $E_f$  为

$$E_f := \{e \in E \mid F(e) \geq F_{E_f}^*(e)\} \quad (5-12)$$

其中， $F_{E_f}^*(e)$  是下述优化问题的极小值：

$$\begin{aligned} \min_{u_{\text{loc}} \in U} F_{E_f}(e) &= q(e) + F(f_e(e, u_{\text{loc}})) \\ \text{s.t.} \quad & f_e(e, u_{\text{loc}}) \in E_f \end{aligned} \quad (5-13)$$

拟采用第 2 章中用子集序列逼近终端状态约束集的方法，假设一个包含原点的终端状态约束集是已知的，记为  $E_f^0$ 。基于  $E_f^0$ ，可以构造出另一个终端状态约束集  $E_f^1$ 。

$$E_f^1 := \{e \in E \mid F(e) \geq F_{E_f^0}^*(e)\}$$

其中,  $F_{E_f^0}^*(e)$  是下述优化问题的极小值:

$$\begin{aligned} \min_{u_{\text{loc}} \in U} F_{E_f^0}(e) &= q(e) + F(f_e(e, u_{\text{loc}})) \\ \text{s.t.} \quad & f_e(e, u_{\text{loc}}) \in E_f^0 \end{aligned}$$

类似地,  $E_f^2$  可以通过相同的方法构造出来。

$$E_f^2 := \{e \in E \mid F(e) \geq F_{E_f^1}^*(e)\}$$

其中,  $F_{E_f^1}^*(e)$  是下述优化问题的极小值:

$$\begin{aligned} \min_{u_{\text{loc}} \in U} F_{E_f^1}(e) &= q(e) + F(f_e(e, u_{\text{loc}})) \\ \text{s.t.} \quad & f_e(e, u_{\text{loc}}) \in E_f^1 \end{aligned}$$

如此反复,  $E_f^j, j = 3, 4, \dots, \infty$  都可以被构造出来。

$$E_f^j := \{e \in E \mid F(e) \geq F_{E_f^{j-1}}^*(e)\}$$

其中,  $F_{E_f^{j-1}}^*(e)$  是下述优化问题的极小值:

$$\begin{aligned} \min_{u_{\text{loc}} \in U} F_{E_f^{j-1}}(e) &= q(e) + F(f_e(e, u_{\text{loc}})) \\ \text{s.t.} \quad & f_e(e, u_{\text{loc}}) \in E_f^{j-1} \end{aligned}$$

与第 2 章类似, 采用支持向量机对  $E_f^j$  进行估计。同样, 可设置一个终止迭代阈值, 得到  $E_f$  的最终估计。具体方法见第 2 章, 此处不再详述。

## 5.2.5 鲁棒预测控制器设计

考虑干扰输入输出系统为

$$y_{k+1} = f(y_k, u_k) + w_k \quad (5-14)$$

其中,  $w_k \in W$  为有界系统干扰。

定义  $y_d$  为给定的稳定期望输出,  $\bar{u}$  为使得  $0 = f_e(0, \bar{u})$ , 即  $y_d = f(y_d, \bar{u})$  成立的控制输入。类似地, 将系统的输入输出模型式 (5-14) 转换为跟踪误差的动态过程模型

$$e_{k+1} = f_e(e_k, \Delta u_k + \bar{u}) = f_{e\bar{u}}(e_k, \Delta u_k) + w_k \quad (5-15)$$

其中,  $e_k = y_k - y_d$ ,  $\Delta u_k = u_k - \bar{u}$ ,  $f_{e\bar{u}}(e_k, \Delta u_k) = f(e_k + y_d, \Delta u_k + \bar{u})$ 。

此时, 系统式 (5-14) 对  $(y_k, u_k) = (y_d, \bar{u})$  的逼近就转变为系统式 (5-15) 对  $(e_k, \Delta u_k) = (0, 0)$  的镇定, 可直接采用第 4 章中介绍的鲁棒预测控制方法, 此处不再详述。

### 5.2.6 仿真算例

考虑如下输入输出系统:

$$\begin{aligned} y_1(k+1) &= y_1(k) + T_s(y_2(k) + w_1(k)) \\ y_2(k+1) &= y_2(k) + T_s(-k_0 \times y_1(k) \times (1 + k_d \times y_1^2(k)) - c \times y_1^2(k) + u(k) + w_2(k)) \end{aligned}$$

系统中各参数的具体数值为:  $k_0 = 0.3$ ;  $k_d = 0.7$ ;  $c = 0.2$ ; 采样时间  $T_s = 0.1s$ ; 输入约束为  $-0.5 \leq u \leq 0.5$ ; 输出约束为  $-0.1 \leq y_2 \leq 0.5$ ; 系统干扰分别为  $w_1(k) = \text{rand}(-0.0025, 0.0025)$ ,  $w_2(k) = \text{rand}(-0.0025, 0.0025)$ , 其中,  $\text{rand}(-0.0025, 0.0025)$  表示  $(-0.0025, 0.0025)$  区间上的平均分布随机噪声。

取预测步长  $N = 20$ , 阶段指标函数为  $q(y - y_d) = (y - y_d)^T I_{2 \times 2} (y - y_d)$ , 控制目标设置为系统输出由  $(0.2, 0.05)$  出发, 依次穿越  $y_d = (0.3, 0)$ ,  $y_d = (0.4, 0)$ ,  $y_d = (0.7, 0)$ , 最后到达  $y_d = (0.5, 0)$ 。分别求得  $\bar{u} = 0.1137$ ,  $\bar{u} = 0.1654$ ,  $\bar{u} = 0.38$ ,  $\bar{u} = 0.2263$  为针对这 4 个期望输出点满足  $y_d = f(y_d, \bar{u})$  的控制输入, 将系统的跟踪问题分别围绕这 4 对  $(y_d, \bar{u})$  转换为镇定问题, 采用 4.3 节介绍的鲁棒预测控制方法得到如图 5-1 所示的仿真结果。

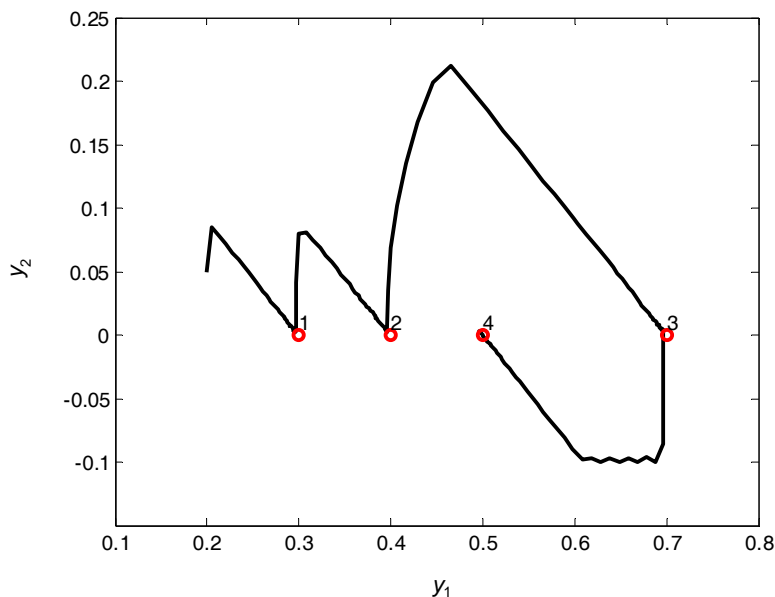


图 5-1 输出的跟踪曲线

在图 5-1 中，标号为“1”“2”“3”“4”的小圆圈表示系统输出需要依次穿越的期望点，实线表示系统的输出曲线。可以看出，采用本节介绍的预测控制方法可使系统输出跟踪事先设置好的期望输出点。

## 5.3 对期望输出轨迹的跟踪

### 5.3.1 问题描述

这种形式的跟踪问题指的是要求系统的输出跟踪某一条事先设置好的期望输出轨迹。假设期望输出轨迹为  $Y_{\text{ref}} = \{y_{\text{ref}}(0), y_{\text{ref}}(1), \dots, y_{\text{ref}}(\infty)\}$ ，本节要解决的问题可陈述为：

对于给定的期望输出轨迹  $Y_{\text{ref}}$ ，设计预测控制器，使得系统输出能跟踪这条期望输出轨迹。

### 5.3.2 稳定性分析

初始时刻, 预测控制的优化问题可描述为

$$\begin{aligned}
 \min_{u(i, y_0) \in U} J(u, y_0) &= \sum_{i=0}^{N-1} q((y(i, y_0) - y_{\text{ref}}(i)), u(i, y_0)) + \\
 &\quad F(y(N, y_0) - y_{\text{ref}}(N)) \\
 \text{s.t. } y(i+1, y_0) &= f(y(i, y_0), u(i, y_0)) \\
 y(i+1, y_0) &\in Y, u(i, y_0) \in U, i=0, \dots, N-1 \\
 y(N, y_0) &\in Y_f(y_{\text{ref}}(N))
 \end{aligned} \tag{5-16}$$

其中,  $y_{\text{ref}}(i)$  表示  $i$  时刻的期望输出,  $Y_f(y_{\text{ref}}(N))$  为输出的终端状态约束集, 它与  $N$  时刻的期望输出  $y_{\text{ref}}(N)$  有关。考虑到阶段指标函数取形式为  $q(y - y_{\text{ref}}, u) = (y - y_{\text{ref}})^T Q(y - y_{\text{ref}})$ , 优化问题式 (5-16) 可简化为

$$\begin{aligned}
 \min_{u(i, y_0) \in U} J(u, y_0) &= \sum_{i=0}^{N-1} q((y(i, y_0) - y_{\text{ref}}(i))) + F(y(N, y_0) - y_{\text{ref}}(N)) \\
 \text{s.t. } y(i+1, y_0) &= f(y(i, y_0), u(i, y_0)) \\
 y(i+1, y_0) &\in Y, u(i, y_0) \in U, i=0, \dots, N-1 \\
 y(N, y_0) &\in Y_f(y_{\text{ref}}(N))
 \end{aligned} \tag{5-17}$$

同样, 定义跟踪误差  $e$  为系统输出与期望输出的差值, 即  $e = y - y_{\text{ref}}$ 。将跟踪误差代入系统的输入输出模型式 (5-3), 可得跟踪误差的动态过程为

$$e_{k+1} + y_{\text{ref}}(k+1) = f(e_k + y_{\text{ref}}(k), u_k) \tag{5-18}$$

定义

$$f_k(e_k, u_k) = f(e_k + y_{\text{ref}}(k), u_k) - y_{\text{ref}}(k+1) \tag{5-19}$$

则动态过程式 (5-18) 可以转换为

$$e_{k+1} = f_k(e_k, u_k) \tag{5-20}$$

类似地, 由于  $f(\cdot, \cdot)$  为已知非线性函数, 而  $y_{\text{ref}}(k)$ 、 $y_{\text{ref}}(k+1)$  为已知的期望输出, 因此,  $f_k(\cdot, \cdot)$  也是已知的。此时, 优化问题式 (5-17) 可转换为

$$\begin{aligned} \min_{u(i, e_0) \in U} J(u, e_0) &= \sum_{i=0}^{N-1} q(e(i, e_0)) + F(e(N, e_0)) \\ \text{s.t. } e(i+1, e_0) &= f_i(e(i, e_0), u(i, e_0)) \\ e(i+1, e_0) + y_{\text{ref}}(i+1) &\in Y, \quad u(i, e_0) \in U, \quad i=0, \dots, N-1 \\ e(N, e_0) &\in E_f \end{aligned} \quad (5-21)$$

其中,  $E_f$  为跟踪误差的终端状态约束集。

考虑如下假设。

**假设 5.3** 存在原点 (此处的原点意指  $e=0$ ) 的一个邻域  $E_f$ , 满足:

(TC1)  $E_f$  为包含零点的闭集, 并且对任意的  $e \in E_f$ 、任意的  $y_{\text{ref}}(k) \in Y_{\text{ref}}$ , 都有  $e + y_{\text{ref}}(k) \in Y$ 。

(TC2) 对任意的  $e \in E_f$ 、任意的  $f_k(\cdot, \cdot)$ , 都存在  $u_{\text{loc}, k} \in U$ , 使得  $F(e) \geq q(e) + F(f_k(e, u_{\text{loc}, k}))$  成立, 并且  $f_k(e, u_{\text{loc}, k}) \in E_f$ 。

**定理 5.2** 当假设 5.3 成立时, 如果初始跟踪误差  $e_0$  满足  $e_0 + y_{\text{ref}}(0) \in Y$  且  $e^*(N, e_0) \in E_f$ , 则采用滚动控制律  $u_{\text{RH}}$  可使  $e_0$  最终收敛到零。

证明与定理 5.1 的证明类似。

假设  $k=0$  时刻的最优输入序列为  $u^*(e_0) = \{u^*(0, e_0), \dots, u^*(N-1, e_0)\}$ , 最优跟踪误差序列为  $e^*(e_0) = \{e^*(1, e_0), \dots, e^*(N, e_0)\}$ 。

将  $u^*(0, e_0)$  作用于系统, 过渡到下一时刻, 跟踪误差为  $e_1 = f_0(e_0, u^*(0, e_0))$ , 构造输入序列  $u(e_1) = \{u^*(1, e_0), u^*(2, e_0), \dots, u^*(N-1, e_0), u_{\text{loc}, N}\}$ 。其中,  $u_{\text{loc}, N}$  为满足  $F(e^*(N, e_0)) \geq q(e^*(N, e_0)) + F(f_N(e^*(N, e_0), u_{\text{loc}, N}))$  并且  $f_N(e^*(N, e_0), u_{\text{loc}, N}) \in E_f$  的控制输入。

显然,  $\mathbf{u}(e_1)$  是系统在  $k=1$  时刻的一组可行控制律。而与  $\mathbf{u}(e_1)$  对应的性能指标  $J(\mathbf{u}(e_1), e_1)$  满足

$$\begin{aligned} J(\mathbf{u}(e_1), e_1) - J^*(e_0) &= q(e^*(N, e_0)) + F(f_N(e^*(N, e_0), u_{\text{loc}, N})) - \\ &\quad q(e_0) - F(e^*(N, e_0)) \\ &\leq -q(e_0) \\ &\leq 0 \end{aligned}$$

所以有

$$J^*(e_1) - J^*(x_0) \leq J(\mathbf{u}(e_1), e_1) - J^*(x_0) \leq 0$$

证毕。

从上述稳定性分析中可以看出, 假设 5.3 中对跟踪误差的终端状态约束集  $E_f$  的要求非常苛刻, 有时甚至会出现终端状态约束集只能取 0 或根本不存在的状况。举个一维情况下的简单例子, 输出  $y$  被限幅在  $[-5, 5]$  之内,  $k=1$  时刻的期望输出为  $y_{\text{ref}}(1) = -5$ ,  $k=2$  时刻的期望输出为  $y_{\text{ref}}(2) = 5$ , 此时, 终端状态约束集就只能取  $E_f = 0$ 。如果某一个  $y_{\text{ref}}(k)$  是不稳定的, 那么, 终端状态约束集就退化到空集。另外, 就算存在这样一个满足假设 5.3 的  $E_f$ , 其求取也是非常困难的, 因为不同的  $k$  时刻, 期望输出  $y_{\text{ref}}(k)$  是不同的, 与其对应的  $f_k(\cdot, \cdot)$  也是不同的。如果期望输出跨幅很大, 则求取  $E_f$  时计算量是非常大的, 有时甚至根本就不可求。

### 5.3.3 预测控制器设计

为避免对  $E_f$  的求取带来的困难, 我们引入大致跟踪的概念, 放宽对跟踪误差的收敛性要求。

**定义 5.1** 对于给定的跟踪误差范围  $E_r = \{e \mid \|e\|_{\infty} \leq \varepsilon\}$ , 如果系统的跟踪

误差落入此范围之内，即  $y - y_{\text{ref}} \in E_r$ ，则称系统输出能大致跟踪期望输出。

由定义 5.1 可以看出，系统输出能大致跟踪期望输出指的是每一时刻的跟踪误差都在给定的范围之内。因此，预测控制的优化问题中应当考虑优化初始时刻的下一时刻的跟踪误差约束，而不是跟踪误差的终端状态约束，这是因为通过预测控制的优化问题求解的最优控制序列只有第一项真正作用于实际系统。

此时，关于跟踪误差的滚动优化问题可描述为

$$\begin{aligned} \min_{u(i, e_0) \in U} J(u, e_0) &= \sum_{i=0}^{N-1} q(e(i, e_0)) + F(e(N, e_0)) \\ \text{s.t. } e(i+1, e_0) &= f_i(e(i, e_0), u(i, e_0)) \\ e(i+1, e_0) + y_{\text{ref}}(i+1) &\in Y, u(i, e_0) \in U, i = 0, \dots, N-1 \\ e(1, e_0) &\in E_r \end{aligned} \quad (5-22)$$

将关于跟踪误差的优化问题切换至关于输出的优化问题，可得初始时刻关于输出的优化问题为

$$\begin{aligned} \min_{u(i, y_0) \in U} J(u, y_0) &= \sum_{i=0}^{N-1} q(y(i, y_0) - y_{\text{ref}}(i)) + F(y(N, y_0) - y_{\text{ref}}(N)) \\ \text{s.t. } y(i+1, y_0) &= f(y(i, y_0), u(i, y_0)) \\ y(i+1, y_0) &\in Y, u(i, y_0) \in U, i = 0, \dots, N-1 \\ y(1, y_0) - y_{\text{ref}}(1) &\in E_r \end{aligned} \quad (5-23)$$

$k$  时刻关于输出的优化问题为

$$\begin{aligned} \min_{u(i, y_k) \in U} J(u, y_k) &= \sum_{i=0}^{N-1} q(y(i, y_k) - y_{\text{ref}}(i+k)) + \\ &\quad F(y(N, y_k) - y_{\text{ref}}(N+k)) \\ \text{s.t. } y(i+1, y_k) &= f(y(i, y_k), u(i, y_k)) \\ y(i+1, y_k) &\in Y, u(i, y_k) \in U, i = 0, \dots, N-1 \\ y(1, y_k) - y_{\text{ref}}(1+k) &\in E_r \end{aligned} \quad (5-24)$$



由优化问题式 (5-24) 可以看出, 对预测输出的约束只有当前时刻的下一时刻才有。考虑到在跟踪问题中, 要求每一时刻的预测输出都尽量逼近期望输出, 而不是对优化末端时刻的预测输出有更强的逼近要求, 因此可以不用在终端代价函数中对末端输出误差施加额外的权重比例。考虑到阶段指标函数的形式为  $q(y - y_{\text{ref}}) = (y - y_{\text{ref}})^T Q(y - y_{\text{ref}})$ , 可取终端代价函数的形式为

$$F(y - y_{\text{ref}}) = (y - y_{\text{ref}})^T Q(y - y_{\text{ref}}) \quad (5-25)$$

又考虑到  $y(0, y_k) = y_k$  是系统在  $k$  时刻的实际输出,  $q(y(0, y_k) - y_{\text{ref}}(k))$  已经是一个固定值, 与当前时刻及未来时刻的输入没有关系, 所以, 优化问题式 (5-24) 可以用更简洁的形式描述, 如下:

$$\begin{aligned} \min_{u(i, y_k) \in U} J(u, y_k) &= \sum_{i=0}^{N-1} q(y(i+1, y_k) - y_{\text{ref}}(i+1+k)) \\ \text{s.t. } y(i+1, y_k) &= f(y(i, y_k), u(i, y_k)) \\ y(i+1, y_k) &\in Y, u(i, y_k) \in U, i=0, \dots, N-1 \\ y(1, y_k) - y_{\text{ref}}(1+k) &\in E_r \end{aligned} \quad (5-26)$$

此时, 系统的大致跟踪可用预测控制的优化问题来进行描述。假设当前时刻为  $k$ , 如果优化问题式 (5-26) 有可行解, 则称系统输出在  $k+1$  时刻是能大致跟踪期望输出  $y_{\text{ref}}(1+k)$  的。

**假设 5.4** 对于任意的  $y_{\text{ref}}(k) \in Y_{\text{ref}}$ 、任意的  $e \in E_r$ , 都存在  $u(y_{\text{ref}}(k), e) \in U$ , 使得  $f(y_{\text{ref}}(k) + e, u(y_{\text{ref}}(k), e)) - y_{\text{ref}}(k+1) \in E_r$ 。

**定理 5.3** 当假设 5.4 成立时, 如果初始时刻的系统输出满足  $y_0 - y_{\text{ref}}(0) \in E_r$ , 则采用以优化问题式 (5-26) 描述的预测控制器能使系统输出在跟踪误差范围为  $E_r$  时大致跟踪期望输出轨迹  $Y_{\text{ref}}$ 。

证明：

假设当前时刻为  $k$ ，优化问题式 (5-26) 有解。定义优化问题式 (5-26) 的最优控制序列为  $u(i, y_k), i = 0, 1, 2, \dots, N-1$ ，将  $u(0, y_k)$  作用于系统，系统过渡到  $k+1$  时刻。因为式 (5-26) 有解，则意味着  $k+1$  时刻的实际输出  $y_{k+1}$  满足

$$y_{k+1} - y_{\text{ref}}(1+k) \in E_r$$

即

$$e_{k+1} \in E_r$$

又由假设 5.4 可知，存在

$$u(y_{\text{ref}}(k+1), e_{k+1}) \in U$$

使得

$$f(y_{k+1}, u(y_{\text{ref}}(k+1), e_{k+1})) - y_{\text{ref}}(k+2) \in E_r$$

成立。

因此， $k+1$  时刻对应的优化问题有解。

又因为初始时刻的系统输出满足  $y_0 - y_{\text{ref}}(0) \in E_r$ ，因此初始时刻对应的优化问题有解。

所以，系统输出能大致跟踪期望输出轨迹  $Y_{\text{ref}}$ 。

得证。

### 5.3.4 鲁棒预测控制器设计

本节针对干扰输入输出系统式 (5-14)

$$y_{k+1} = f(y_k, u_k) + w_k$$

设计能使系统输出大致跟踪期望输出轨迹的鲁棒预测控制器。其设计思路与无干扰系统类似，但是需要考虑系统干扰对跟踪性能的影响。

当系统存在干扰时，预测控制在  $k$  时刻的优化问题可描述为

$$\begin{aligned} \min_{u(i, y_k) \in U} \max_{w_i \in W} J(u, y_k) &= \sum_{i=0}^{N-1} q(y(i+1, y_k) - y_{\text{ref}}(i+1+k)) \\ \text{s.t. } y(i+1, y_k) &= f(y(i, y_k), u(i, y_k)) + w_i \\ y(i+1, y_k) &\in Y, u(i, y_k) \in U, i = 0, \dots, N-1 \\ y(1, y_k) - y_{\text{ref}}(1+k) &\in E_r \end{aligned} \quad (5-27)$$

**假设 5.5** 对任意的  $y_{\text{ref}}(k) \in Y_{\text{ref}}$ 、任意的  $e \in E_r$  及任意的  $w \in W$ ，都存在  $u(y_{\text{ref}}(k), e) \in U$ ，使得  $f(y_{\text{ref}}(k) + e, u(y_{\text{ref}}(k), e)) + w - y_{\text{ref}}(k+1) \in E_r$ 。

可以看出，假设 5.5 相对于假设 5.4 来说，要求更苛刻一些，因为干扰的引入使得系统的稳定性降低。此时，为保证系统输出能大致跟踪期望输出轨迹，就需要施加额外的条件。而且系统跟踪误差的范围也要适当进行调整，如果  $E_r$  选得太小，例如， $E_r \subset W$ ，那么，以  $E_r$  为跟踪误差范围定义的大致跟踪，系统是根本达不到的。所以，系统跟踪误差范围的选取必须满足  $W \subseteq E_r$ 。

**定理 5.4** 当假设 5.5 成立时，如果初始时刻的系统输出满足  $y_0 - y_{\text{ref}}(0) \in E_r$ ，则采用以优化问题式 (5-27) 描述的鲁棒预测控制器能使系统输出在跟踪误差范围为  $E_r$  时大致跟踪期望输出轨迹  $Y_{\text{ref}}$ 。

定理 5.4 的证明过程与定理 5.3 的证明类似，此处不再详述。

### 5.3.5 仿真算例

考虑如下仿真对象：

$$y_1(k+1) = y_1(k) + T_s \left( \frac{-0.2y_1(k)}{1+y_1^2(k)} + \sin(u_1(k)) + 1.5u_2(k) \right) + w_1(k)$$

$$y_2(k+1) = y_2(k) + T_s \left( 0.2u_1(k) + \frac{1.5u_2(k)}{1+u_2^2(k)} - 0.6y_2(k) \right) + w_2(k)$$

在该对象中, 采样时间为  $T_s = 0.1s$ , 系统干扰范围为  $W = \{(w_1, w_2) \mid |w_1| \leq 0.01, |w_2| \leq 0.01\}$ ; 输入约束为  $U = \{(u_1, u_2) \mid |u_1| \leq 1, |u_2| \leq 1\}$ ; 输出约束为  $Y = \{(y_1, y_2) \mid |y_1| \leq 2, |y_2| \leq 2\}$ 。

本次仿真分三种情况。

## 1. 第一种情况

期望输出轨迹设置为

$$y_{1,\text{ref}}(k) = 0.5 \times (1 - \sin(k \times \pi / 20))$$

$$y_{2,\text{ref}}(k) = 0.5 \times (1 - \cos(k \times \pi / 20)), \quad k = 0, 1, 2, \dots, \infty$$

跟踪误差范围设置为

$$E_r = \left\{ (e_{y_1}, e_{y_2}) \left| \begin{array}{l} e_{y_1} = |y_1 - y_{1,\text{ref}}| \leq 0.005 \\ e_{y_2} = |y_2 - y_{2,\text{ref}}| \leq 0.005 \end{array} \right. \right\}$$

在这种情况下, 程序在求解第一次优化时就中止, 因为跟踪误差范围比系统的干扰范围还小, 优化问题式 (5-27) 中的约束条件无法得到满足。

## 2. 第二种情况

期望输出轨迹设置为

$$y_{1,\text{ref}}(k) = 0.5 \times (1 - \sin(k \times \pi / 20))$$

$$y_{2,\text{ref}}(k) = 0.5 \times (1 - \cos(k \times \pi / 20))$$

跟踪误差范围设置为

$$E_r = \left\{ (e_{y_1}, e_{y_2}) \left| \begin{array}{l} e_{y_1} = |y_1 - y_{1,\text{ref}}| \leq 0.03 \\ e_{y_2} = |y_2 - y_{2,\text{ref}}| \leq 0.03 \end{array} \right. \right\}$$

仿真结果如图 5-2~图 5-5 所示。

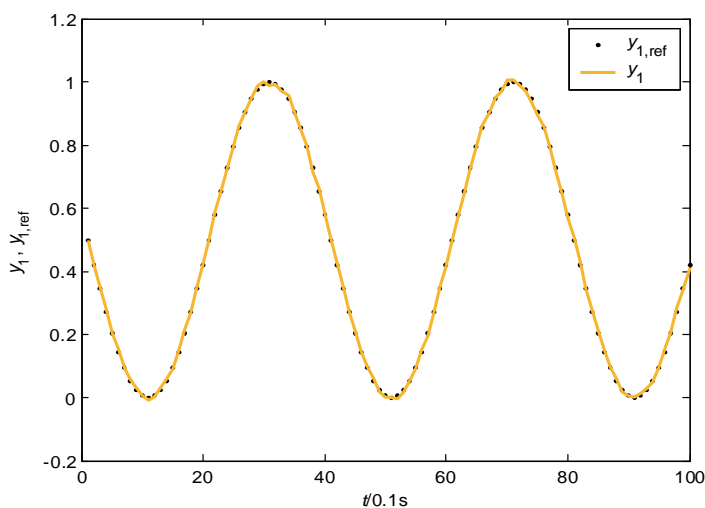


图 5-2  $y_1$  的跟踪曲线

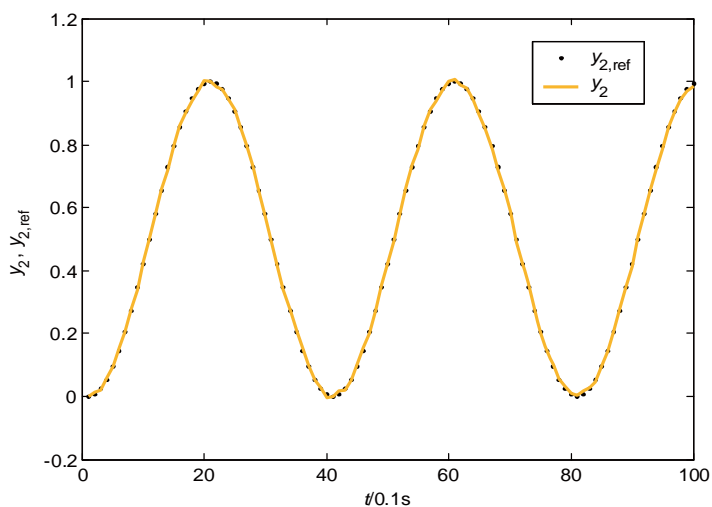


图 5-3  $y_2$  的跟踪曲线

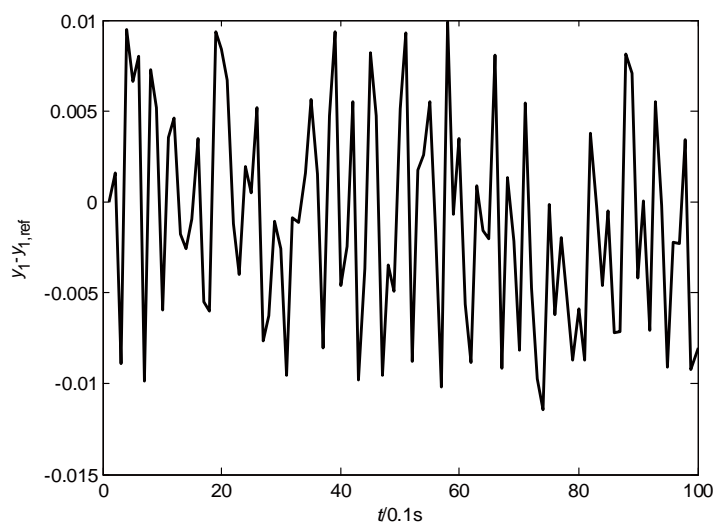


图 5-4  $y_1$  的跟踪误差

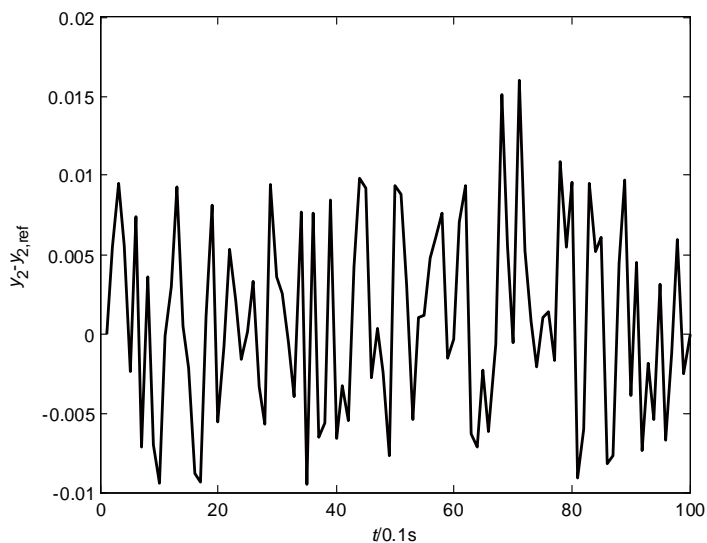


图 5-5  $y_2$  的跟踪误差

### 3. 第三种情况

期望输出轨迹设置为

$$y_{1,\text{ref}}(k) = 1 \times (1 - \sin(k \times \pi / 20))$$

$$y_{2,\text{ref}}(k) = 1 \times (1 - \cos(k \times \pi / 20))$$

跟踪误差范围设置为

$$E_r = \left\{ (e_{y_1}, e_{y_2}) \left| \begin{array}{l} e_{y_1} = |y_1 - y_{1,\text{ref}}| \leq 0.03 \\ e_{y_2} = |y_2 - y_{2,\text{ref}}| \leq 0.03 \end{array} \right. \right\}$$

仿真结果如图 5-6 和图 5-7 所示。

在第二种情况下，系统输出能在事先设置的跟踪误差范围内大致跟踪期望输出。但是，当对期望输出进行调整后，如第三种情况所示，系统输出大致跟踪期望输出到某一时刻时，就无法再跟踪下去，原因是期望输出的设置与跟踪误差范围的设置无法再满足假设 5.5，优化问题式 (5-27) 没有可行解。

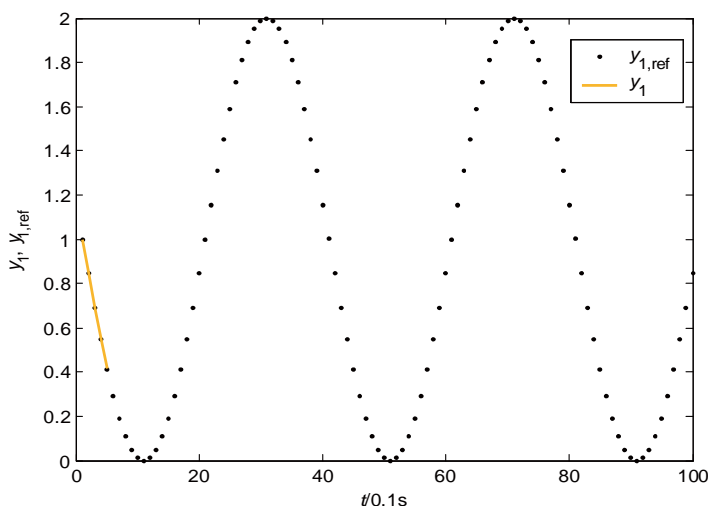
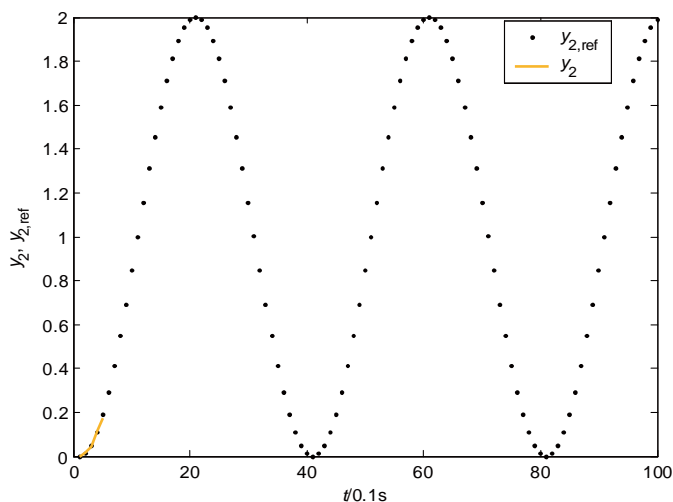


图 5-6  $y_1$  的跟踪曲线

图 5-7  $y_2$  的跟踪曲线

## 5.4 滚动控制律的次优解法

由于非线性优化的计算量都比较大，很多学者采用将非线性模型线性化的方法来简化优化问题的求解。例如，文献[100]利用中值定理将非线性系统进行线性化处理，用三次样条函数逼近线性系统的时变参数，并对样条函数的常值参数进行辨识；文献[101]采取直接辨识控制器参数的方法构建直接广义预测控制器，但是对辨识参数的初值选取很敏感。文献[102]针对只含有输出非线性环节（系统方程中与输出相关的非线性项）的系统沿期望输出轨迹用泰勒一阶展开近似线性化，但是其方法无法处理具有输入非线性环节的系统。本节沿袭将非线性模型线性化的思路，针对一类不可逆非线性系统给出了一种求取跟踪问题中次优预测控制律的方法，思路为：首先求取一条逼近最优滚动控制序列的近似滚动控制序列，然后将非线性系统沿近似滚动控制序列和期望输出轨迹处理为线性系统。鉴于基于斯特林插值公式的级数展开不需要作求导运算<sup>[61-63]</sup>，将斯特林插值公式作为对非线性系统进行线性化处理的工具。



### 5.4.1 基于斯特林插值公式的近似处理

考虑向量  $\mathbf{z}, \bar{\mathbf{z}} \in R^{n_z}$ ，非线性函数  $f: R^{n_z} \rightarrow R$ 。定义  $\tilde{\mathbf{z}} = \mathbf{z} - \bar{\mathbf{z}} = [\tilde{z}_1 \ \tilde{z}_2 \ \cdots \ \tilde{z}_{n_z}]^T$ ，并令  $\tilde{z}_j$  为  $\tilde{\mathbf{z}}$  的第  $j$  个元素， $\mathbf{e}_j$  为第  $j$  个元素为 1 的单位向量，基于斯特林插值公式的一阶展开可描述为<sup>[64,65]</sup>

$$f(\mathbf{z}) \approx f(\bar{\mathbf{z}}) + \sum_{j=1}^{n_z} f_{\text{D1}}^j(\bar{\mathbf{z}}) \tilde{z}_j = f(\bar{\mathbf{z}}) + f_{\text{D1}}(\bar{\mathbf{z}}) \tilde{\mathbf{z}} \quad (5-28)$$

其中， $h$  表示区间长度，为可调参数。

$$f_{\text{D1}}^j(\bar{\mathbf{z}}) = \frac{f(\bar{\mathbf{z}} + h\mathbf{e}_j) - f(\bar{\mathbf{z}} - h\mathbf{e}_j)}{2h} \quad (5-29)$$

$$f_{\text{D1}}(\bar{\mathbf{z}}) = [f_{\text{D1}}^1(\bar{\mathbf{z}}) \ f_{\text{D1}}^2(\bar{\mathbf{z}}) \ \cdots \ f_{\text{D1}}^{n_z}(\bar{\mathbf{z}})] \quad (5-30)$$

用斯特林插值公式对非线性函数进行展开与泰勒级数展开类似，当  $h \rightarrow 0$  时，式 (5-29) 就是泰勒展开中的求导运算。但是，正是由于不要求  $h \rightarrow 0$ ，使得它可处理并不严格满足连续可导条件的非线性系统。

### 5.4.2 近似滚动控制序列的求取

考虑如下不可逆非线性输入输出系统：

$$y(k) = f(y(k-1), y(k-2), \cdots, y(k-n_a), u(k-1), u(k-2), \cdots, u(k-n_b)) \quad (5-31)$$

其中， $k$  为正整数； $y(\cdot)$  为系统输出， $n_a$  为输出阶次； $u(\cdot)$  为系统输入， $n_b$  为输入阶次； $f(\cdot, \cdots, \cdot)$  为关于  $y(k-1), \cdots, y(k-n_a), u(k-1), \cdots, u(k-n_b)$  的已知非线性函数，满足斯特林有界； $Y$  和  $U$  分别表示系统的输出约束集和控制（输入）约束集。

期望输出轨迹设置为  $Y_{\text{ref}} = \{y_{\text{ref}}(0), y_{\text{ref}}(1), \cdots, y_{\text{ref}}(\infty)\}$ ，考虑预测控制在  $k$  时刻的优化问题为

$$\begin{aligned}
 \min_{u(k+i)} J(\mathbf{u}, y(k)) &= \sum_{i=0}^{N-1} q(y(k+i+1) - y_{\text{ref}}(k+i+1)) \\
 \text{s.t. } y(k+i+1) &= f(y(k+i), \dots, y(k+i+1-n_a), u(k+i), \dots, \\
 &\quad u(k+i+1-n_b)) \\
 y(k+i+1) &\in Y, u(k+i) \in U, i=0, \dots, N-1 \\
 y(k+1) - y_{\text{ref}}(k+1) &\in E_r
 \end{aligned} \tag{5-32}$$

**假设 5.6** 对于期望输出轨迹  $Y_{\text{ref}}$  和给定的误差范围  $E_r$ ，采用以优化问题式 (5-32) 描述的预测控制的滚动控制律，系统输出能大致跟踪期望输出轨迹。

当假设 5.6 成立时，定义  $\mathbf{u}_{\text{RH}} = \{u_{\text{RH}}(0), u_{\text{RH}}(1), \dots, u_{\text{RH}}(\infty)\}$  为采用以优化问题式 (5-32) 描述的预测控制的滚动控制序列。在没有对系统实施真正的控制之前， $\mathbf{u}_{\text{RH}}$  是未知的。此时，可用一条近似滚动控制序列来逼近  $\mathbf{u}_{\text{RH}}$ ，记为

$$\mathbf{u}_r = \{u_r(0), u_r(1), \dots, u_r(\infty)\}$$

以下内容给出了一种求取近似滚动控制序列的方法。设当前时刻为  $k$ ，近似滚动控制  $u_r(\cdot)$  的选取分  $k-j$  时刻和  $k+j$  时刻两种情况。

在  $k-j, j=1, 2, \dots, k$  时刻，近似滚动控制取系统实际输入。

$$u_r(k-j) = u(k-j), i=1, 2, \dots \tag{5-33}$$

在  $k+j, j=0, 1, \dots, \infty$  时刻，按如下方式选取：

$$u_r(k) = \begin{cases} u_{\min}, & u_r(k-1) + \Delta u_r(k) < u_{\min} \\ u_r(k-1) + \Delta u_r(k), & u_{\min} \leq u_r(k-1) + \Delta u_r(k) \leq u_{\max} \\ u_{\max}, & u_r(k-1) + \Delta u_r(k) > u_{\max} \end{cases} \tag{5-34}$$

其中， $u_{\min}, u_{\max}$  表示控制输入幅值约束的上下界。

$$\Delta u_r(k) = \frac{\left( y_{\text{ref}}(k+1) - f \left( \begin{array}{c} y_{\text{ref}}(k), y_{\text{ref}}(k-1), \dots, y_{\text{ref}}(k+1-n_a), \\ u_r(k-1), u_r(k-1), \dots, u_r(k+1-n_b) \end{array} \right) \right)}{D_u^1(k+1)} \quad (5-35)$$

$$D_u^1(k+1) = \frac{f \left( \begin{array}{c} y_{\text{ref}}(k), y_{\text{ref}}(k-1), \dots, y_{\text{ref}}(k+1-n_a), \\ u_r(k-1) + h, u_r(k-1), \dots, u_r(k+1-n_b) \end{array} \right)}{2h} - \frac{f \left( \begin{array}{c} y_{\text{ref}}(k), y_{\text{ref}}(k-1), \dots, y_{\text{ref}}(k+1-n_a), \\ u_r(k-1) - h, u_r(k-1), \dots, u_r(k+1-n_b) \end{array} \right)}{2h} \quad (5-36)$$

**注 5.1** 求取  $k$  时刻近似滚动控制的思路为：先用已知的  $u_r(k-1)$  近似替代未知的  $u_r(k)$ ，预测  $k+1$  时刻的输出，再由此预测输出与  $y_{\text{ref}}(k+1)$  的差值来修正  $u_r(k-1)$  得到最终的  $u_r(k)$ 。

**注 5.2** 当选定  $h$  后，式 (5-36) 中用到的都是已知量（包括期望输出和  $k$  时刻之前的近似滚动控制）；式 (5-35) 中有两个  $u_r(k-1)$ ，第一个  $u_r(k-1)$  只是  $u_r(k)$  的近似替代；通过式 (5-34) 求取最终的  $u_r(k)$  用到的都是已知量。

同理， $u_r(k+1)$  可由已经计算出来的  $u_r(k)$  近似替代，再用  $y_{\text{ref}}(k+2)$  修正得到最终的  $u_r(k+1)$ 。依次类推，可得

$$u_r(k+j) = \begin{cases} u_{\min}, & u_r(k+j-1) + \Delta u_r(k+j) < u_{\min} \\ u_r(k+j-1) + \Delta u_r(k+j), & u_{\min} \leq u_r(k+j-1) + \Delta u_r(k+j) \leq u_{\max} \\ u_{\max}, & u_r(k+j-1) + \Delta u_r(k+j) > u_{\max} \end{cases} \quad (5-37)$$

其中

$$\Delta u_r(k+j) = \frac{\left( y_{\text{ref}}(k+j+1) - f \left( \begin{array}{c} y_{\text{ref}}(k+j), y_{\text{ref}}(k+j-1), \dots, y_{\text{ref}}(k+j+1-n_a), \\ u_r(k+j-1), u_r(k+j-1), \dots, u_r(k+j+1-n_b) \end{array} \right) \right)}{D_u^1(k+j+1)} \quad (5-38)$$

$$\begin{aligned}
 & D_u^1(k+j+1) \\
 &= \frac{f\left(y_{\text{ref}}(k+j), y_{\text{ref}}(k+j-1), \dots, y_{\text{ref}}(k+j+1-n_a), \right. \\
 &\quad \left. u_r(k+j-1)+h, u_r(k+j-1), \dots, u_r(k+j+1-n_b)\right)}{2h} - \\
 &\quad \frac{f\left(y_{\text{ref}}(k+j), y_{\text{ref}}(k+j-1), \dots, y_{\text{ref}}(k+j+1-n_a), \right. \\
 &\quad \left. u_r(k+j-1)-h, u_r(k+j-1), \dots, u_r(k+j+1-n_b)\right)}{2h} \quad (5-39)
 \end{aligned}$$

### 5.4.3 预测输出的求取

求得近似滚动控制序列后,即可将非线性系统沿近似滚动控制序列和期望输出轨迹处理为线性系统。为方便起见,首先定义实际控制作用与近似滚动控制作用之间的差值为修正输入  $du$ 。

当前时刻为  $k$ ,显然,  $k-j$  时刻的近似滚动控制取系统实际输入,因此

$$du(k-j)=0, j=0,1,2,\dots \quad (5-40)$$

而  $k+j$  时刻的修正输入为

$$du(k+j)=u(k+j)-u_r(k+j) \quad (5-41)$$

可以看出,使性能指标  $J$  取得极小值的  $u(\cdot)$  可通过优化  $du(\cdot)$  间接得到其次优值。在  $k$  时刻,将  $y(k+1)$  沿期望输出  $y_{\text{ref}}(k), y_{\text{ref}}(k-1), \dots, y_{\text{ref}}(k-n_a+1)$  和近似滚动控制序列  $u_r(k), u_r(k-1), \dots, u_r(k-n_b+1)$  用斯特林插植公式一阶展开,可得

$$\begin{aligned}
 y(k+1) &\approx \hat{y}(k+1) \\
 &= f\left(y_{\text{ref}}(k), \dots, y_{\text{ref}}(k+1-n_a), \right. \\
 &\quad \left. u_r(k), \dots, u_r(k+1-n_b)\right) + \\
 &\quad \sum_{i=1}^{n_a} D_{yi}(k+1) \times (y(k+1-i) - y_{\text{ref}}(k+1-i)) + \\
 &\quad \sum_{i=1}^{n_b} D_{ui}(k+1) \times du(k+1-i) \quad (5-42)
 \end{aligned}$$

考虑到

$$du(k-1) = du(k-2) = \cdots = du(k+1-n_b) = 0$$

式(5-42)可以简化为

$$\begin{aligned} \hat{y}(k+1) = & f\left(y_{\text{ref}}(k), \cdots, y_{\text{ref}}(k+1-n_a), \right. \\ & \left. u_r(k), \cdots, u_r(k+1-n_b)\right) + \\ & \sum_{i=1}^{n_a} D_{yi}(k+1) \times (y(k+1-i) - y_{\text{ref}}(k+1-i)) + \\ & D_{u1}(k+1) \times du(k) \end{aligned} \quad (5-43)$$

同理有

$$\begin{aligned} \hat{y}(k+j) = & f\left(y_{\text{ref}}(k+j-1), \cdots, y_{\text{ref}}(k+j-n_a), \right. \\ & \left. u_r(k+j-1), \cdots, u_r(k+j-n_b)\right) + \\ & \sum_{i=1}^{n_a} D_{yi}(k+j) \times (\hat{y}(k+j-i) - y_{\text{ref}}(k+j-i)) + \\ & D_{u1}(k+j) \times du(k+j-1) + \\ & D_{u2}(k+j) \times du(k+j-2) + \cdots + \\ & D_{uj}(k+j) \times du(k) \end{aligned} \quad (5-44)$$

其中, 当  $j-i \leq 0$  时,  $\hat{y}(k+j-i)$  表示当前时刻之前(包括当前时刻)的输出近似值, 可直接取系统的实际输出  $\hat{y}(k+j-i) = y(k+j-i)$ 。  $D_{yi}(k+j)$ ,  $i=1, 2, \cdots, n_a$  和  $D_{ui}(k+j)$ ,  $i=1, 2, \cdots, j$  均为已知量, 其求取方法如下:

$$\begin{aligned} D_{yi}(k+j) = & \frac{f\left(y_{\text{ref}}(k+j-1), \cdots, y_{\text{ref}}(k+j-i)+h, \cdots, y_{\text{ref}}(k+j-n_a), \right. \\ & \left. u_r(k+j-1), \cdots, u_r(k+j-n_b)\right) - \\ & f\left(y_{\text{ref}}(k+j-1), \cdots, y_{\text{ref}}(k+j-i)-h, \cdots, y_{\text{ref}}(k+j-n_a), \right. \\ & \left. u_r(k+j-1), \cdots, u_r(k+j-n_b)\right)}{2h} \end{aligned} \quad (5-45)$$

$$D_{ui}(k+j) = \frac{f\left(\begin{matrix} y_{\text{ref}}(k+j-1), \dots, y_{\text{ref}}(k+j-n_a), \\ u_r(k+j-1), \dots, u_r(k+j-i)+h, \dots, u_r(k+j-n_b) \end{matrix}\right)}{2h} - \frac{f\left(\begin{matrix} y_{\text{ref}}(k+j-1), \dots, y_{\text{ref}}(k+j-n_a), \\ u_r(k+j-1), \dots, u_r(k+j-i)-h, \dots, u_r(k+j-n_b) \end{matrix}\right)}{2h} \quad (5-46)$$

为方便起见, 将式 (5-44) 简记为

$$\begin{aligned} \hat{y}(k+j) = f_{k+j} & \left( \hat{y}(k+j-1), \dots, \hat{y}(k+j-n_a), \right. \\ & \left. du(k+j-1), \dots, du(k), 0, \dots, 0 \right) \end{aligned} \quad (5-47)$$

其中,  $f_{k+j}(\cdot, \dots, \cdot)$  为关于  $\hat{y}(k+j-1), \dots, \hat{y}(k+j-n_a), du(k+j-1), \dots, du(k), 0, \dots, 0$  的线性函数; 当  $j-i \leq 0, i=1, 2, \dots, n_a$  时,  $\hat{y}(k+j-i) = y(k+j-i)$ 。

#### 5.4.4 次优控制律的求取

经过上述处理后, 预测控制的优化问题式 (5-32) 可转换为

$$\begin{aligned} \min_{du(k+i)} J(u, y(k)) &= \sum_{i=0}^{N-1} q(\hat{y}(k+i+1) - y_{\text{ref}}(k+i+1)) \\ \text{s.t. } \hat{y}(k+i+1) &= f_{k+i+1} \left( \hat{y}(k+i), \dots, \hat{y}(k+1), y(k), \dots, y(k+i+1-n_a), \right. \\ & \quad \left. du(k+i), \dots, du(k), 0, \dots, 0 \right) \quad (5-48) \\ \hat{y}(k+i+1) &\in Y, du(k+i) + u_r(k+i) \in U, i=0, \dots, N-1 \\ \hat{y}(k+1) - y_{\text{ref}}(k+1) &\in E_r \end{aligned}$$

求解式 (5-48), 可得到一个修正控制序列  $\{du(k+i)\}$ , 将其与近似滚动控制序列  $\{u_r(k+i)\}$  相加, 即可得到一个原非线性预测控制的次优控制序列。在实际控制中, 只有  $u(k) = du(k) + u_r(k)$  作用于系统, 下一时刻的控制由下一时刻的优化给出, 如此反复, 得到一个次优滚动控制序列。

将非线性模型进行线性近似处理后, 基于线性化模型的在线优化计算量要比直接基于非线性模型的在线优化计算量小得多。但是, 线性化截断误差

的引入使得系统的稳定性受到影响, 为了保证稳定性, 对系统的要求就更加苛刻。

**假设 5.7** 存在一个线性化截断误差范围  $\hat{E}$ , 考虑  $E_{\hat{r}} = \{a+b | a \in E_r, b \in \hat{E}\}$ , 对于任意的  $y_{\text{ref}}(k), \dots, y_{\text{ref}}(k+n_a-1) \in Y_{\text{ref}}$ 、任意的  $e(k), \dots, e(k+n_a-1) \in E_{\hat{r}}$ 、任意的  $u(k+n_a-1), \dots, u(k+n_a-n_b) \in U$ , 以及满足  $u(k+n_a-1) + du(k+n_a-1) \in U$  的  $du(k+n_a-1)$ , 都有

$$\begin{aligned} & y(k+n_a) - \hat{y}(k+n_a) \\ &= f \left( \begin{array}{c} y_{\text{ref}}(k+n_a-1) + e(k+n_a-1), \dots, y_{\text{ref}}(k) + e(k) \\ u(k+n_a-1) + du(k+n_a-1), u(k+n_a-2), \dots, u(k+n_a-n_b) \end{array} \right) - \\ & \quad f_{k+n_a} \left( y_{\text{ref}}(k+n_a-1) + e(k+n_a-1), \dots, y_{\text{ref}}(k) + e(k), du(k+n_a-1), \dots, 0 \right) \\ & \in \hat{E} \end{aligned}$$

其中,  $f_{k+n_a}(\cdot, \dots, \cdot)$  为用斯特林插值公式对  $f(\cdot, \dots, \cdot)$  的逼近。

**假设 5.8** 对于任意的  $y_{\text{ref}}(k), \dots, y_{\text{ref}}(k+n_a-1) \in Y_{\text{ref}}$ 、任意的  $e(k), \dots, e(k+n_a-1) \in E_{\hat{r}}$ , 都存在满足  $u_r(k+n_a-1) + du(k+n_a-1) \in U$  的  $du(k+n_a-1)$ , 使得

$$\begin{aligned} & \hat{y}(k+n_a) - y_{\text{ref}}(k+n_a) \\ &= f_{k+n_a} \left( y_{\text{ref}}(k+n_a-1) + e(k+n_a-1), \dots, y_{\text{ref}}(k) + e(k), du(k+n_a-1), \dots, 0 \right) \\ & \in E_r \end{aligned}$$

其中,  $u_r(k+n_a-1)$  为采用式 (5-34) 求取的近似滚动控制。

**定理 5.5** 当假设 5.7 和假设 5.8 都成立时, 如果初始时刻的系统输出满足  $y(0) - y_{\text{ref}}(0) \in E_{\hat{r}}$ , 则采用上述方法求解得到的次优滚动控制律可使系统输出在跟踪误差范围  $E_{\hat{r}}$  内大致跟踪期望输出轨迹。

证明:

假设当前时刻为  $k$ , 优化问题式 (5-48) 有解, 定义其最优修正输入序

列为  $\{du(k+i), i=0,1,\dots,N-1\}$ ，将  $u(k)=du(k)+u_r(k)$  作用于系统，系统过渡到  $k+1$  时刻。因为式 (5-48) 有解，则意味着  $\hat{y}(k+1)$  满足

$$\hat{y}(k+1)-y_{\text{ref}}(k+1)\in E_r$$

又由假设 5.7 可知， $k+1$  时刻的实际输出  $y(k+1)$  满足

$$y(k+1)-\hat{y}(k+1)\in \hat{E}$$

因此

$$y(k+1)-y_{\text{ref}}(k+1)\in E_{\hat{r}}$$

这意味着，存在一个  $e(k+1)\in E_{\hat{r}}$ ，使得  $y(k+1)=y_{\text{ref}}(k+1)+e(k+1)$ 。

由假设 5.8 可知，存在修正输入  $du(k+1)$ ，使得

$$\hat{y}(k+2)-y_{\text{ref}}(k+2)\in E_r$$

因此， $k+1$  时刻对应的优化问题有解。

又因为初始时刻的系统实际输出满足  $y(0)-y_{\text{ref}}(0)\in E_{\hat{r}}$ ，因此初始时刻对应的优化问题有解。所以，系统输出能大致跟踪期望输出轨迹。

得证。

由定理 5.5 可以看出，线性化截断误差的引入使得系统输出的跟踪精度降低，原滚动控制律可使系统输出在跟踪误差范围  $E_r$  内大致跟踪期望输出，而次优滚动控制律只能使系统输出在跟踪误差范围  $E_{\hat{r}}$  内大致跟踪期望输出。

### 5.4.5 仿真算例

本次仿真取被控对象为



$$y(k+1) = \begin{cases} y(k) + 0.1 \left( 2 \sin(y(k-1)^2) + 5u(k) - y(k)u(k-1) \right), & y(k) > 1 \\ y(k) + 0.1 \left( 4 \cos(y(k-1)^2) + 7u(k) - 2y(k)u(k-1) \right), & y(k) \leq 1 \end{cases}$$

系统控制输入的幅值约束为

$$-2 \leq u \leq 2$$

考虑期望输出为

$$y_{\text{ref}}(k) = 5 \sin\left(\frac{\pi(k-1)}{20}\right), k=1, 2, \dots$$

预测步长取  $N=3$ ，可调参数取  $h=0.5$ ，初始参数设置为

$$y_{\text{ref}}(-1) = y_{\text{ref}}(0) = 0; y(-1) = y(0) = 0; u(-1) = u(0) = 0$$

基于原非线性模型的大致跟踪误差范围取  $-0.6 \leq e \leq 0.6$ ，经过多次仿真验证得出，基于线性近似模型的大致跟踪误差范围可取  $-1 \leq e \leq 1$ 。

图 5-8 和图 5-9 是仿真 100 步的结果。

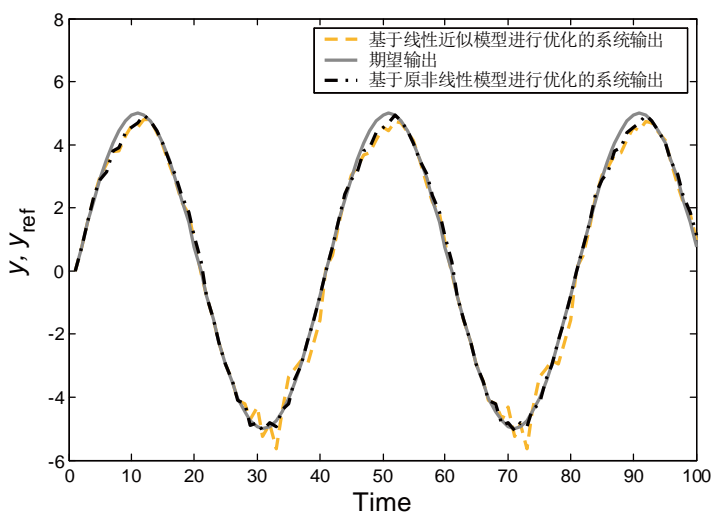


图 5-8 输出跟踪对比

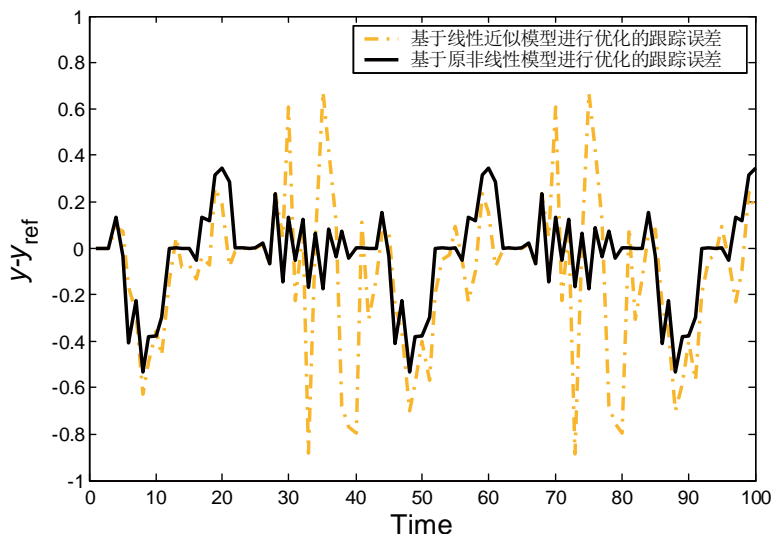


图 5-9 跟踪误差对比

分别将仿真 100 步后两种方法进行优化计算的时间消耗进行统计求均值，得到两种方法平均每步优化计算的消耗时间。

经计算，基于近似线性模型的优化计算消耗时间为 0.02491s；直接基于原非线性模型的优化计算消耗时间为 0.05899s。

可以看出，将基于近似线性模型进行优化得到的控制律作用于系统，其跟踪精度要比将直接基于原非线性模型进行优化得到的控制律作用于系统的跟踪精度低一些。其原因是在对非线性模型进行线性近似的处理过程中，不可避免地引入了线性化截断误差，使得系统的跟踪精度降低。但是，基于近似线性模型进行优化的平均时间消耗要比直接基于原非线性模型进行优化的平均时间消耗要少得多。

## 5.5 基于线性近似和神经网络逼近的非线性系统预测控制

非线性系统预测控制的关键问题在于如何处理非线性和降低在线计算量。由于非线性系统的设计和分析往往比线性系统困难得多,所以非线性微分方程通常不可能得到封闭形式的解析解,除非经过特殊处理。而傅里叶变换、拉普拉斯变换、叠加原理等适用于线性系统的数学工具都不再适用于非线性系统的控制问题。因此,对于实际工程人员和理论研究者来说,处理非线性系统最直接也是最有效的办法就是将系统的非线性特性线性化,然后再用成熟的线性系统设计方法完成控制过程<sup>[119]</sup>。

在 5.4 节中,我们针对不同类型的非线性系统,分别用泰勒级数和斯特林插值公式进行了线性近似处理,并设计了模型预测控制器。在进行预测控制器设计时,为了减少在线计算量,人为地忽略了线性化过程中产生的非线性高阶项。如果能对被忽略的非线性高阶项进行有效估计,则无疑会大大提高预测模型的精度,从而改善控制效果。

由于神经网络(Neural Network, NN)能够充分逼近复杂的非线性映射关系,已成为对非线性系统建立预测模型和进行优化控制的关键技术之一<sup>[120]</sup>,并产生了多种基于神经网络的预测控制算法<sup>[121-127]</sup>。当前,反向传播(Back Propagation, BP)网络是应用最为成熟的神经网络,但存在局部最优、训练速度慢等缺点;而径向基函数(Radial Basis Function, RBF)神经网络在一定程度上克服了 BP 网络的缺点,已在预测控制领域得到较好的应用<sup>[128,129]</sup>;文献[130]以二次型函数作为预测控制器的目标函数,使用递归 BP 神经网络构造一个非线性多步预测模型,设计了一种非线性多步预测控制器;文献[131]采用多层局部回归神经网络离线建立预测模型,采用二次型函数作为性能指标函数,设计了预测控制器,改善了控制性能。

受文献[130]和文献[131]的启发,本节设计了基于线性近似和二次型重构的非线性模型预测控制器。通过将对象非线性系统进行线性近似处理,并将控制性能指标函数重构为易于处理的二次型问题,明显减少了在线计算量,同时得到了较满意的控制效果。在此基础上,为了提高模型精确度,用RBF神经网络对线性化过程中产生的非线性高阶项进行逼近,显著提高了对象系统模型的精确度,控制效果也得到了进一步改善。

### 5.5.1 问题描述

考虑如下的离散时间非线性系统:

$$\begin{aligned} x(k+1) &= f(x(k), u(k)) \\ y(k) &= C(k)x(k) \end{aligned} \quad (5-49)$$

其中,  $x(k) \in R^n$  为状态向量,  $u(k) \in R^m$  为控制输入向量,  $y(k) \in R^p$  为输出向量。 $f(\cdot)$  为一个连续的非线性函数,  $C(k)$  为一个常数矩阵。

在实际的工业生产过程中,受设备功率等条件的限制,对系统的控制量和控制增量都是有一定限制的,需服从如下的约束条件:

$$\begin{aligned} u_{\min} &\leq u(k) \leq u_{\max} \\ \Delta u_{\min} &\leq \Delta u(k) \leq \Delta u_{\max} \end{aligned} \quad (5-50)$$

同时,为了避免系统输出超过限制量,给出以下的输出约束:

$$y_{\min} \leq y(k) \leq y_{\max} \quad (5-51)$$

其中,  $u_{\min} \leq u_{\max}$ 、 $\Delta u_{\min} \leq \Delta u_{\max}$  和  $y_{\min} \leq y_{\max}$  分别为各向量的下界和上界。

## 5.5.2 基于线性近似的非线性模型预测控制

### 1. 对象系统的线性近似

由于线性化是处理非线性系统最直接也最有效的方法,因此近年来发展出了若干以微分几何为主要工具的精确线性化方法,为非线性控制系统的分析与综合问题提供了有效的解决方案。精确线性化方法在应用时必须满足苛刻的条件,并且结构相对复杂,在实际的应用过程中往往很难得到所需的非线性变换结果。另外,大多数实际系统并不满足精确线性化方法中的对合条件,因而研究非线性系统的近似处理方法受到了工程技术人员和理论研究者的高度重视。近年来出现的先进的线性近似方法主要包括最佳拟线性化、伪线性化、近似输入输出线性化、近似状态反馈线性化、中心流形法等<sup>[119]</sup>。但在工程应用领域,传统的泰勒级数展开法仍然受到工程技术人员的青睐,这是因为泰勒级数展开法具有物理意义明确、线性近似过程简单的优点。而斯特林插值公式法<sup>[135]</sup>在现有的关于线性近似的文献中并不多见,但文献<sup>[132]</sup>指出利用斯特林插值公式对非线性函数进行线性化处理的精度要高于泰勒一阶近似,且不需要对其进行求导运算。因此,本节拟用上述两种方法对预测控制中的非线性系统进行线性近似处理。

在本节中,将根据非线性函数 $f(\cdot)$ 是否可微,分别用泰勒级数展开法和斯特林插值公式法对系统进行不同的线性近似处理。当然,斯特林插值公式法也可对连续可微的非线性函数进行线性近似,在这里之所以采用泰勒级数展开法对其进行处理,是因为泰勒级数展开法有其自身优点,即物理意义明确、线性近似过程简单。

#### 1) 输入输出函数可微分的情况

当 $f(\cdot)$ 为一个连续可微的非线性函数时,可直接用泰勒级数公式将非线性系统式(5-49)在工作点 $(x_0, u_0)$ 处展开,分解为一个线性系统和一个未知非线性项相加的形式。

$$\begin{aligned} x(k+1) = & f(x_0, u_0) + \frac{\partial f(x, u)}{\partial x} \Big|_{x_0, u_0} (x(k) - x_0) + \\ & \frac{\partial f(x, u)}{\partial u} \Big|_{x_0, u_0} (u(k) - u_0) + \varepsilon(k) \end{aligned} \quad (5-52)$$

其中,  $f(x_0, u_0) = x_0$ , 而  $\varepsilon(k)$  是相应的泰勒级数的高阶项。定义  $A(k) = \frac{\partial f(x, u)}{\partial x} \Big|_{x_0, u_0}$ ,  $B(k) = \frac{\partial f(x, u)}{\partial u} \Big|_{x_0, u_0}$ ,  $\sigma x(k) = x(k) - x_0$  和  $\sigma u(k) = u(k) - u_0$ , 则系统式 (5-49) 可重写为

$$\begin{aligned} \sigma x(k+1) &= A(k) \sigma x(k) + B(k) \sigma u(k) + \varepsilon(k) \\ \sigma y(k) &= C(k) \sigma x(k) \end{aligned} \quad (5-53)$$

## 2) 输入输出函数不可微的情况

当  $f(\cdot)$  为一个连续不可微的非线性函数时, 泰勒级数在此不再适用。与泰勒级数展开线性近似不同, 用斯特林插值公式对非线性函数进行线性近似时无须进行求导运算, 只需用有限线性运算来逼近该非线性函数<sup>[133,134]</sup>。由于斯特林插值公式能对余项直接进行控制, 这使得在某些应用中其比泰勒近似更具吸引力。如果能选取合适的区间长度, 则可以确保斯特林余项能接近完整泰勒级数的高阶项<sup>[135]</sup>。

下面用斯特林插值公式对非线性系统式 (5-49) 进行线性近似。首先定义如下函数:

$$\begin{aligned} \delta f(x) &= f\left(x + \frac{h}{2}\right) - f\left(x - \frac{h}{2}\right) \\ \mu f(x) &= \frac{f\left(x + \frac{h}{2}\right) + f\left(x - \frac{h}{2}\right)}{2} \end{aligned} \quad (5-54)$$

其中,  $f(\cdot): R \rightarrow R$  为非线性函数;  $h$  为一个可调参数, 一般为取自区间 (0,1) 内的实数;  $\delta$  为差分算子;  $\mu$  为平均算子。令  $ph = \tilde{x} = x - \bar{x}$ , 用斯特林插值公式将  $f(x)$  在  $x = \bar{x}$  处展开, 表述如下:

$$\begin{aligned}
f(x) &= f(\bar{x} + ph) \\
&= f(\bar{x}) + p\mu\delta f(\bar{x}) + \frac{p^2}{2!}\delta^2 f(\bar{x}) + \dots
\end{aligned} \tag{5-55}$$

此处只保留其一阶项，将高阶项写为一个参数的形式，则非线性函数的线性近似为

$$\begin{aligned}
f(x) &= f(\bar{x}) + f_{D1}(\bar{x})ph + \omega(k) \\
&= f(\bar{x}) + f_{D1}(\bar{x})\tilde{x} + \omega(k)
\end{aligned} \tag{5-56}$$

其中

$$f_{D1}(\bar{x}) = \frac{f(\bar{x} + h) - f(\bar{x} - h)}{2h} \tag{5-57}$$

拓展到向量形式，即  $x \in R^n$ ,  $f: R^n \rightarrow R^n$ ,  $\tilde{x} = [\tilde{x}_1 \quad \tilde{x}_2 \quad \dots \quad \tilde{x}_n]^T$ 。令  $\tilde{x}_j$  为  $\tilde{x}$  的第  $j$  个元素， $e_j$  为第  $j$  个元素为1的单位向量，则式 (5-56) 可拓展为

$$\begin{aligned}
f(x) &= f(\bar{x}) + \tilde{D}_{\sigma x} f + \frac{1}{2!} \tilde{D}_{\sigma x}^2 f + \dots \\
&= f(\bar{x}) + \tilde{D}_{\sigma x} f + \omega(k)
\end{aligned} \tag{5-58}$$

其中

$$\tilde{D}_{\sigma x} f = \frac{1}{h} \left( \sum_{i=1}^n \Delta x_i \mu_i \delta_i \right) f(\bar{x}) \tag{5-59}$$

式 (5-54) 拓展为

$$\begin{aligned}
\delta_i f(\bar{x}) &= f\left(\bar{x} + \frac{h}{2} e_i\right) - f\left(\bar{x} - \frac{h}{2} e_i\right) \\
\mu_i f(\bar{x}) &= \frac{f\left(\bar{x} + \frac{h}{2} e_i\right) + f\left(\bar{x} - \frac{h}{2} e_i\right)}{2}
\end{aligned} \tag{5-60}$$

其中， $e_i$  为第  $i$  个元素为1的单位向量。

下面用斯特林插值公式将非线性系统式(5-49)在工作点 $(x_0, u_0)$ 处展开。此时,  $\bar{x} = [x_0 \quad u_0]^T, x = [x(k) \quad u(k)]^T, n = 2$ 。则式(5-58)重写为

$$\begin{aligned} x(k+1) &= f(x(k), u(k)) \\ &= \frac{1}{h} (\mu_1 \delta_1 (x(k) - x_0) + \mu_2 \delta_2 (u(k) - u_0)) f(x_0, u_0) + \\ &\quad f(x_0, u_0) + \omega(k) \end{aligned} \quad (5-61)$$

其中,  $f(x_0, u_0) = x_0$ , 而 $\omega(k)$ 是相应的斯特林插值公式的高阶项。差分算子 $\delta_1, \delta_2$ 和平均算子 $\mu_1, \mu_2$ 可由式(5-60)计算得出。定义 $A'(k) = \frac{\delta_1 \mu_1 x_0}{h}$ ,  $B'(k) = \frac{\delta_2 \mu_2 x_0}{h}$ ,  $\sigma x(k) = x(k) - x_0$ 和 $\sigma u(k) = u(k) - u_0$ , 则系统式(5-53)可重写为

$$\begin{aligned} \sigma x(k+1) &= A'(k) \sigma x(k) + B'(k) \sigma u(k) + \omega(k) \\ \sigma y(k) &= C(k) \sigma x(k) \end{aligned} \quad (5-62)$$

至此, 无论非线性函数 $f(\cdot)$ 是否可微, 非线性系统式(5-49)均可以通过线性近似方法构造为如式(5-53)或式(5-62)的形式。这种形式可为构造二次型最优化问题提供方便, 将在下面详细说明。

## 2. 性能指标函数的二次型重构

在第1章中已经提到, 模型预测控制是一种迭代最优化方法, 即在每个采样时刻 $k$ 测量或估计当前状态, 然后通过对一个控制性能指标函数的最优化来获得最优的控制输入向量。对模型式(5-49), 给出如下满足有限域二次型规则的性能指标函数:

$$J(k) = \sum_{j=1}^N \|r(k+j|k) - y(k+j|k)\|_Q^2 + \sum_{j=0}^{N_u-1} \|\Delta u(k+j|k)\|_R^2 \quad (5-63)$$

其中,  $r(k+j|k)$ 为输出信号的参考轨迹,  $y(k+j|k)$ 为预测输出,  $\Delta u(k+j|k)$ 为输入增量, 且 $\Delta u(k+j|k) = u(k+j|k) - u(k-1+j|k)$ 。  $N$ 和 $N_u$ 分别为预测



域和控制域, 其中  $N \geq 1$ ,  $0 < N_u \leq N$ 。  $\mathbf{Q}$  和  $\mathbf{R}$  为适当选取的权值矩阵。 $\|\cdot\|_Q$  和  $\|\cdot\|_R$  为权值欧氏范数。式 (5-63) 等号右边第一项表示预测输出和参考输出之间的误差, 第二项考虑了控制增量的影响。在点  $(x_0, u_0)$  处,  $y_0(k) = r(k)$ , 因此性能指标函数式 (5-63) 可以重写为

$$J(k) = \sum_{j=1}^N \|\sigma y(k+j|k)\|_Q^2 + \sum_{j=0}^{N_u-1} \|\Delta u(k+j|k)\|_R^2 \quad (5-64)$$

定义如下向量:

$$\begin{aligned} \bar{\varepsilon}(k) &= [\varepsilon(k|k) \quad \cdots \quad \varepsilon(k+N-1|k)]^T \\ \bar{\omega}(k) &= [\omega(k|k) \quad \cdots \quad \omega(k+N-1|k)]^T \\ \sigma \bar{y}(k) &= [\sigma y(k+1|k) \quad \cdots \quad \sigma y(k+N|k)]^T \\ \sigma \bar{u}(k) &= [\sigma u(k|k) \quad \cdots \quad \sigma u(k+N_u-1|k)]^T \\ \sigma \bar{x}(k) &= [\sigma x(k+1|k) \quad \cdots \quad \sigma x(k+N|k)]^T \\ \Delta \bar{u}(k) &= [\Delta u(k|k) \quad \cdots \quad \Delta u(k+N_u-1|k)]^T \end{aligned} \quad (5-65)$$

则根据式 (5-63), 预测输出  $\sigma \bar{y}(k)$  可表示为

$$\sigma \bar{y}(k) = \mathbf{K} \sigma x(k) + \mathbf{L} \sigma u(k-1) + \mathbf{G} \Delta \bar{u}(k) + \mathbf{H} \bar{\varepsilon}(k) \quad (5-66)$$

根据式 (5-62), 预测输出  $\sigma \bar{y}(k)$  可表示为

$$\sigma \bar{y}(k) = \mathbf{K}' \sigma x(k) + \mathbf{L}' \sigma u(k-1) + \mathbf{G}' \Delta \bar{u}(k) + \mathbf{H}' \bar{\omega}(k) \quad (5-67)$$

由于式 (5-66) 和式 (5-67) 都是由线性近似方法得来的, 不仅具有相似的表示形式, 本质上也并无区别, 因此, 基于式 (5-66) 和式 (5-67) 的二次型重构形式也是相似的。下面仅以式 (5-66) 为例进行说明。

式 (5-66) 中各参数分别为

$$\begin{aligned}
 \mathbf{K} &= [\mathbf{CA} \quad \mathbf{CA}^2 \quad \cdots \quad \mathbf{CA}^N]^T \\
 \mathbf{L} &= \begin{bmatrix} \mathbf{CB} \\ \mathbf{C}(\mathbf{A} + \mathbf{I})\mathbf{B} \\ \vdots \\ \mathbf{C}(\mathbf{A}^{N-1} + \cdots + \mathbf{I})\mathbf{B} \end{bmatrix} \\
 \mathbf{G} &= \begin{bmatrix} \mathbf{CB} & \cdots & 0 \\ \mathbf{C}(\mathbf{A} + \mathbf{I})\mathbf{B} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \mathbf{C}(\mathbf{A}^{N-1} + \cdots + \mathbf{I})\mathbf{B} & \cdots & \mathbf{C}(\mathbf{A}^{N-N_u} + \cdots + \mathbf{I})\mathbf{B} \end{bmatrix} \\
 \mathbf{H} &= \begin{bmatrix} \mathbf{C} & 0 & \cdots & 0 \\ \mathbf{CA} & \mathbf{C} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{N-1} & \mathbf{CA}^{N-2} & \cdots & \mathbf{C} \end{bmatrix}
 \end{aligned}$$

因此, 对式 (5-63) 的最优化问题可重写为

$$\begin{aligned}
 \min_{\Delta \bar{u}(k)} & \left\| \mathbf{K} \sigma x(k) + \mathbf{L} \sigma u(k-1) + \mathbf{G} \Delta \bar{u}(k) + \mathbf{H} \bar{\varepsilon}(k) \right\|_Q^2 + \left\| \Delta \bar{u}(k) \right\|_R^2 \\
 \text{s.t.} & \\
 & \Delta \bar{u}_{\min} \leq \Delta \bar{u}(k) \leq \Delta \bar{u}_{\max} \\
 & \sigma \bar{u}_{\min} \leq \sigma \bar{u}(k-1) + \tilde{\mathbf{I}} \Delta \bar{u}(k) \leq \sigma \bar{u}_{\max} \\
 & \sigma \bar{y}_{\min} \leq \mathbf{K} \sigma x(k) + \mathbf{L} \sigma u(k-1) + \mathbf{G} \Delta \bar{u}(k) + \mathbf{H} \bar{\varepsilon}(k) \leq \sigma \bar{y}_{\max}
 \end{aligned} \tag{5-68}$$

其中

$$\begin{aligned}
 \tilde{\mathbf{I}} &= \begin{bmatrix} \mathbf{I} & 0 & \cdots & 0 \\ \mathbf{I} & \mathbf{I} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{I} & \mathbf{I} & \cdots & \mathbf{I} \end{bmatrix} \\
 \sigma u_{\min} &= u_{\min} - u_0 \quad \sigma u_{\max} = u_{\max} - u_0 \\
 \sigma y_{\min} &= y_{\min} - y_0 \quad \sigma y_{\max} = y_{\max} - y_0
 \end{aligned} \tag{5-69}$$

定义  $\eta = \Delta \bar{u}(k) \in R^N$ ，则对式 (5-64) 的最优化问题可重构为一个二次规划 (QP) 问题。

$$\begin{aligned} \min \quad & \frac{1}{2} \eta^T W \eta + \xi^T \eta \\ \text{s.t.} \quad & s \leq E \eta \leq l \end{aligned} \quad (5-70)$$

其中

$$\begin{aligned} W &= 2(G^T Q G + R) \\ \xi &= 2G^T Q (K \sigma x(k) + L \sigma u(k-1) + H \bar{\varepsilon}(k)) \\ E &= \begin{bmatrix} -\tilde{I} & \tilde{I} & -G & G & I \end{bmatrix}^T \\ s &= \begin{bmatrix} -\infty \\ \Delta \bar{u}_{\min} \end{bmatrix} \\ l &= \begin{bmatrix} b \\ \Delta \bar{u}_{\max} \end{bmatrix} \\ b &= \begin{bmatrix} -\sigma \bar{u}_{\min} + \sigma \bar{u}(k-1) \\ \sigma \bar{u}_{\max} - \sigma \bar{u}(k-1) \\ -\sigma \bar{y}_{\min} + K \sigma x(k) + L \sigma u(k-1) + H \bar{\varepsilon}(k) \\ \sigma \bar{y}_{\max} - K \sigma x(k) - L \sigma u(k-1) - H \bar{\varepsilon}(k) \end{bmatrix} \end{aligned}$$

此处，由于  $Q$  和  $R$  为正定阵，矩阵  $W$  也为正定阵，且所有的可行域都是凸的，因此最优化问题式 (5-70) 为一个凸二次规划问题。对二次规划问题式 (5-70) 求解，得到一个在控制域  $N$  上的控制增量向量  $\Delta \bar{u}(k)$ ，从而可计算出  $k$  时刻的最优控制输入序列。

### 3. 仿真分析

由于斯特林插值公式法处理非线性系统的范围更宽，因此在本节中仅对采用此方法进行线性近似的情况进行仿真分析。选取某非线性对象系统，其非线性函数  $f(\cdot)$  不满足连续可微条件，泰勒级数展开法不再适用于此处，故用斯特林插值公式法对其进行线性近似处理。

**例 5-1** 所选非线性对象系统描述如下：

$$x(k+1) = \begin{cases} x(k) + 0.6 \left( 2 \left( x(k-1) \right)^2 + 4u(k) - x(k)u(k-1) \right), & x(k) > 1 \\ x(k) + 0.6 \left( 4 \left( x(k-1) \right)^2 + 7u(k) - 2x(k)u(k-1) \right), & x(k) \leq 1 \end{cases}$$

其中，预测域  $N=10$ ，控制域  $N_u=3$ ，控制时长  $t=100\text{s}$ 。权值矩阵  $\mathbf{Q}=2.5\mathbf{I}$ ， $\mathbf{R}=0.5\mathbf{I}$ 。输入约束为  $-35 \leq u(k) \leq 35$ ，控制增量约束为  $-40 \leq \Delta u(k) \leq 40$ ，输出约束为  $-10 \leq y(k) \leq 10$ 。本例的参考输出轨迹设置为一个方波信号。

本例对直接基于原非线性模型的情况和基于近似线性模型的情况分别进行了仿真，结果如图 5-10~图 5-12 所示。在图 5-10 中， $y_{\text{ref}}$  为参考输出轨迹， $y_1$  为采用基于原非线性模型的 MPC 方案时系统的输出轨迹， $y_2$  为采用基于本章近似线性模型的 MPC 方案时系统的输出轨迹；图 5-11 为两种不同情况下的控制输入曲线，其中， $u_1$  为采用基于原非线性模型的 MPC 方案时的控制输入， $u_2$  为采用基于本章近似线性模型的 MPC 方案时的控制输入；在图 5-12 中， $e_1 = y_1 - y_{\text{ref}}$ ， $e_2 = y_2 - y_{\text{ref}}$ ，分别表示采用两种 MPC 方案时的系统对参考输出轨迹的跟踪误差。

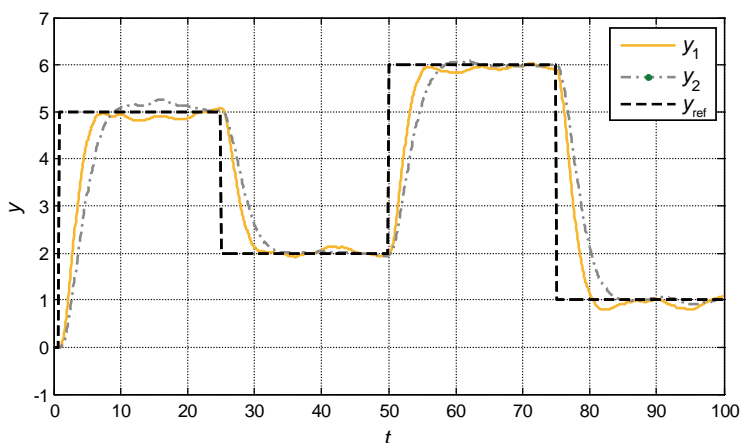


图 5-10 系统输出跟踪曲线

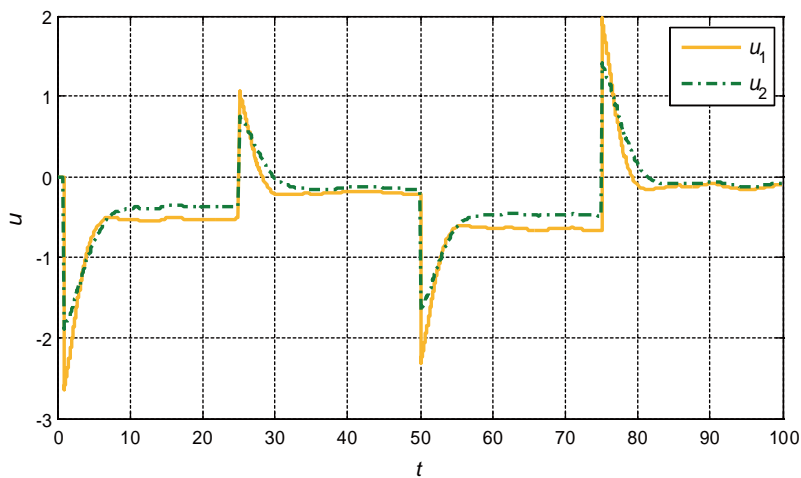


图 5-11 系统控制输入曲线

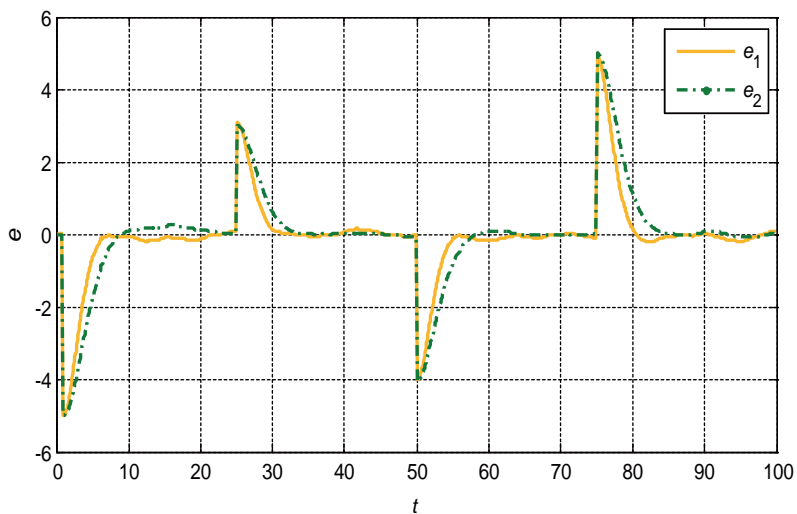


图 5-12 跟踪误差曲线

在仿真过程中，分别对两种方法进行 100 步优化计算所消耗的时间进行统计求均值，可得到两种方法平均每步优化计算所消耗的时间。

经计算，直接基于原非线性模型的 MPC 方案消耗时间为 0.04672s；基于本章近似线性模型的 MPC 方案消耗时间为 0.01925s。

由仿真结果可以看出，本章基于线性近似模型的 MPC 方案的跟踪精度要比直接基于原非线性模型的 MPC 方案的跟踪精度略低。其原因是在对非线性模型进行线性近似处理时，忽略了非线性高阶项，这必然引入了线性化截断误差，使系统的跟踪精度降低。但是，与直接基于原非线性模型的 MPC 方案相比，基于线性近似模型的 MPC 方案能减少控制能量的消耗，而且大大节省了跟踪时间。

### 5.5.3 基于 RBF 神经网络逼近的模型预测控制

#### 1. RBF 神经网络对非线性高阶项的逼近

在上一小节中，我们对非线性对象系统式 (5-49) 进行了线性近似处理，对性能指标函数进行了二次型重构，证明了无论非线性函数  $f(\cdot)$  是否可微，均可以将对式 (5-64) 的最优化问题重写为式 (5-68) 的形式，进而重构为一个凸二次型最优化问题式 (5-70)。但是，仅用线性近似方法并不能得到非线性高阶项  $\varepsilon(k), \omega(k)$  的值，因此，式 (5-65) 中的  $\bar{\varepsilon}(k), \bar{\omega}(k)$  项也未能得出，式 (5-70) 中的参数  $\xi, l$  也为未知。在上一小节的仿真计算中，上述参数都被忽略，从而降低了模型精确度。只有计算或估计出  $\bar{\varepsilon}(k), \bar{\omega}(k)$  的值，才能对二次规划问题式 (5-70) 精确求解，进而准确计算  $k$  时刻的最优控制输入。

RBF 神经网络已经被证明能够以任意精度对某连续函数进行逼近<sup>[136]</sup>。对一个 RBF 神经网络，若网络输入为  $n$  维向量  $u$ ，网络输出为  $m$  维向量  $y$ ，则 RBF 神经网络隐层第  $i$  个节点的输出为

$$q_i = R(\|u - c_i\|) \quad (5-71)$$

其中,  $u$  为  $n$  维输入向量;  $c_i$  为第  $i$  个隐节点的中心,  $i=1,2,\dots$ ;  $\|\cdot\|$  通常为欧氏范数;  $R(\cdot)$  为 RBF 函数, 具有局部感受特性。网络输出层第  $k$  个节点的输出为

$$y_k = \sum_i w_{ki} q_i - \theta_k \quad (5-72)$$

其中,  $w_{ki}$  为  $q_i$  到  $y_k$  的连接权,  $\theta_k$  为第  $k$  个输出节点的阈值。RBF 神经网络的学习算法此处不再赘述, 文献[136]中有详细的训练学习步骤。

用 RBF 神经网络来估计  $\bar{\varepsilon}(k), \bar{\omega}(k)$  值的过程为 (仅以  $\bar{\varepsilon}(k)$  为例):

首先将  $\varepsilon(k)$  重写为

$$\varepsilon(k) = f(x(k), u(k)) - f(x_0, u_0) - A(k)(x(k) - x_0) - B(k)(u(k) - u_0) \quad (5-73)$$

令  $[x \ u \ x_0 \ u_0]^T$  为输入向量,  $\varepsilon(k)$  为目标向量。然后用适当的采样数据对 RBF 神经网络进行训练, 就可以建立目标向量  $\varepsilon(k)$  的模型, 进而可估计出在预测域内每个采样时刻  $k$  的  $\bar{\varepsilon}(k)$  值。

## 2. 基于 RBF 神经网络逼近的 NMPC 方案

结合上面给出的二次规划问题式 (5-70), 基于 RBF 神经网络逼近的 NMPC 方案可总结如下:

(1) 令  $k=1$ 。设置控制时间终点为  $T$ , 预测域为  $N$ , 控制域为  $N_u$ , 采样周期为  $\tau$ , 权值矩阵为  $Q$  和  $R$ 。

(2) 用 RBF 神经网络无导师 (非监督) 学习方法离线建立  $\varepsilon(k)$  模型。

(3) 用第 (2) 步中的训练模型计算  $\bar{\varepsilon}(k)$ 。计算过程模型矩阵  $K, L, G, H, I$  和二次规划矩阵  $W, \xi, E, b$ 。

(4) 对二次规划问题式 (5-70) 求解以获得最优控制增量向量  $\Delta \bar{u}(k)$ 。

(5) 计算最优控制向量  $\bar{u}(k)$ ，仅将第一项控制量  $u(k|k)$  作用于系统。

(6) 如果  $k < T$ ，则令  $k = k + 1$ ，返回第 (3) 步；否则结束。

由以上步骤可以看出，本章提出的算法考虑了线性近似过程中的高阶项  $\bar{\varepsilon}(k), \bar{\omega}(k)$ ，并且可以通过 RBF 神经网络方法对其进行离线估计，不仅能提高模型的精确度，而且可以提高具有高频采样率工业过程的整体效率。

### 3. 仿真分析

本节选取两种不同的非线性对象系统，用上一小节介绍的模型预测控制方案进行仿真分析。例 5-1 中所选取对象系统中的非线性函数  $f(\cdot)$  是可微的，可直接用泰勒级数对其进行线性近似处理；例 5-2 中所选取对象系统中的非线性函数  $f(\cdot)$  是不可微的，用斯特林插值公式法对其进行线性近似处理。本节两个算例均考虑了约束条件的影响，并且都分为考虑高阶项和不考虑高阶项两种情况进行分析，仿真结果表明考虑高阶项时可以达到更好的控制效果。

**例 5-2** 考虑如下非线性被控对象系统：

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 1 & T \\ T & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} T\mu \\ T\mu \end{bmatrix} u_k + \begin{bmatrix} T(1-\mu) & 0 \\ 0 & -4T(1-\mu) \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} u_k$$

其中， $\mu = 0.5$ ，采样时间  $T = 0.1s$ ；预测域  $N = 10$ ，控制域  $N_u = 3$ ，控制时长  $t = 100s$ ；阶段指标函数为  $q(x, u) = x^T Q x + u^T R u$ ，其中， $Q = T \times 0.5I$ ， $R_0 = T \times 1$ ；权值矩阵  $Q = 1.5I$ ， $R = 0.2I$ ；输入约束为  $-25 \leq u(k) \leq 25$ ，控制增量约束为  $-30 \leq \Delta u(k) \leq 30$ ，输出约束为  $-8 \leq y(k) \leq 8$ 。本例的参考输出轨迹设置为一个方波信号。

将对象模型在零点处进行一阶泰勒展开，分解为一个线性系统和一个未知非线性项相加的形式。



$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 1 & 0.1 \\ 0.1 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 0.05 \\ 0.05 \end{bmatrix} u_k + \varepsilon(k)$$

为了应用上一小节介绍的模型预测控制方案,首先需用 RBF 神经网络离线逼近非线性项  $\varepsilon(k)$  的模型。可首先利用高斯白噪声信号作为系统输入,然后用采集到的输入量和输出量来构造训练样本(本仿真中训练样本数据为 800 个),利用 K-means 算法得到隐层各节点的高斯核函数的中心值及宽度向量,在决定好隐层的参数后求出输出层的权值。本例对考虑高阶项的情况和不考虑高阶项的情况分别进行了仿真,结果如图 5-13 和图 5-14 所示。在图 5-13 中,  $y_1$  为考虑高阶项  $\varepsilon(k)$  时系统对参考输出轨迹的跟踪效果,  $y_2$  为不考虑高阶项  $\varepsilon(k)$  时系统对参考输出轨迹的跟踪效果。图 5-14 为两种不同情况下的输入曲线,其中,  $u_1$  为考虑高阶项  $\varepsilon(k)$  时的控制输入,  $u_2$  为不考虑高阶项  $\varepsilon(k)$  时的控制输入。

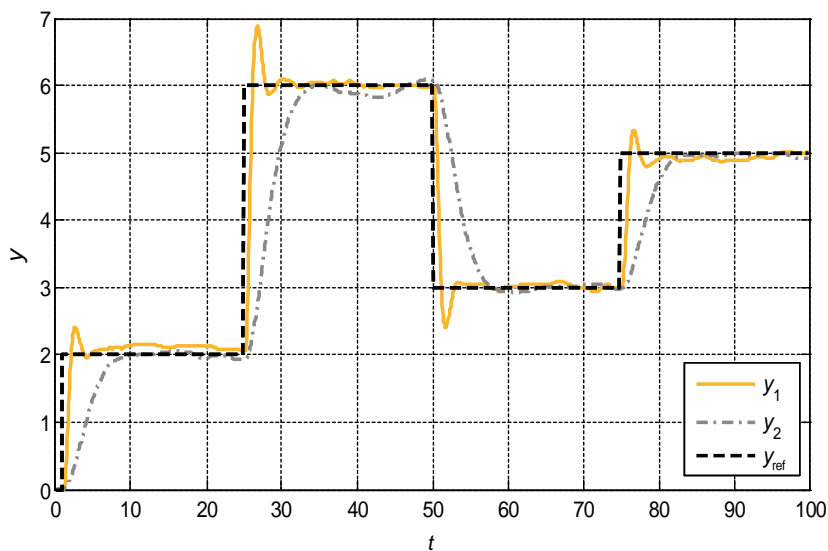


图 5-13 系统输出跟踪曲线

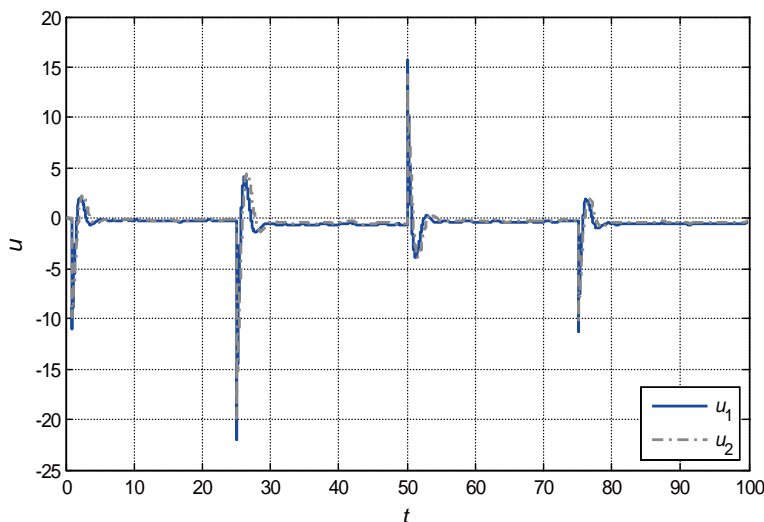


图 5-14 系统控制输入曲线

综合图 5-13 和图 5-14 可以看出,运用本章的 MPC 方案时能对参考输出轨迹进行有效的跟踪,而考虑高阶项时能达到更好的控制效果,在两种情况下的控制输入无显著差别。在仿真结果中,控制输入量、控制增量和输出量均能满足设定的约束条件。

**例 5-3** 本例采用与例 5-1 相同的被控对象系统。

$$x(k+1) = \begin{cases} x(k) + 0.6 \left( 2 \left( x(k-1) \right)^2 + 4u(k) - x(k)u(k-1) \right), & x(k) > 1 \\ x(k) + 0.6 \left( 4 \left( x(k-1) \right)^2 + 7u(k) - 2x(k)u(k-1) \right), & x(k) \leq 1 \end{cases}$$

其中,预测域  $N=10$ ,控制域  $N_u=3$ ,控制时长  $t=100\text{s}$ ;权值矩阵  $\mathbf{Q}=2.5\mathbf{I}$ ,  $\mathbf{R}=0.5\mathbf{I}$ ;输入约束为  $-35 \leq u(k) \leq 35$ ,控制增量约束为  $-40 \leq \Delta u(k) \leq 40$ ,输出约束为  $-10 \leq y(k) \leq 10$ 。本例的参考输出轨迹设置为另一个方波信号。

由于本例的对象系统不满足连续可微的条件,不能用泰勒级数展开法进行处理,故采用斯特林插值公式法对其进行线性近似。参考例 5-1 中的步骤,

本例也对考虑高阶项的情况和不考虑高阶项的情况分别进行了仿真，结果如图 5-15 和图 5-16 所示。在图 5-15 中， $y_1$  为考虑高阶项  $\varepsilon(k)$  时系统对参考输出轨迹的跟踪效果， $y_2$  为不考虑高阶项  $\varepsilon(k)$  时系统对参考输出轨迹的跟踪效果。

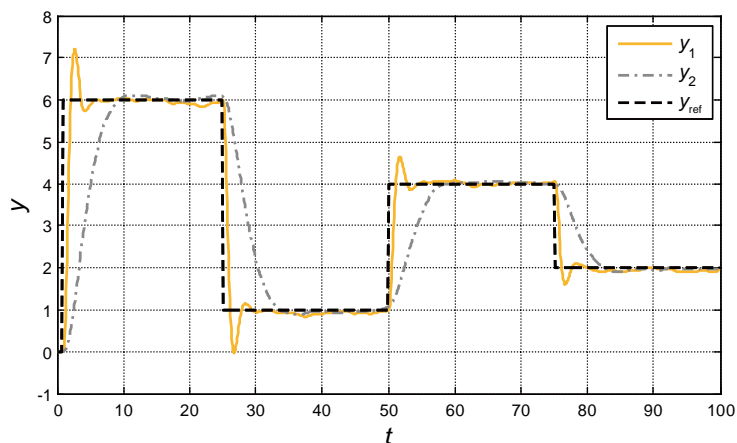


图 5-15 系统输出跟踪曲线

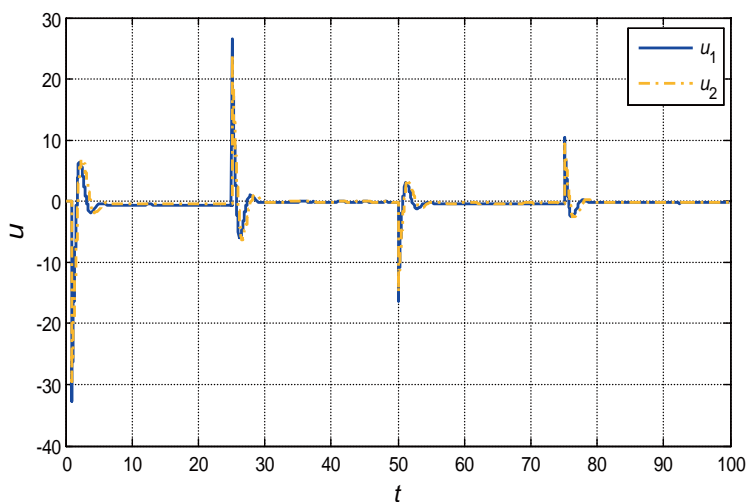


图 5-16 系统控制输入曲线

图 5-16 为两种不同情况下的输入曲线, 其中,  $u_1$  为考虑高阶项  $\varepsilon(k)$  时的控制输入,  $u_2$  为不考虑高阶项  $\varepsilon(k)$  时的控制输入。

综合图 5-15 和图 5-16 可以得出与例 5-2 相似的结论, 两种情况下的控制能量无明显差别, 而考虑高阶项时依然能达到更好的跟踪精度。在仿真结果中, 控制输入量、控制增量和输出量也均能满足约束条件。

## 5.6 本章小结

关于跟踪问题中的非线性预测控制, 现有研究主要集中在设计跟踪常值期望输出的控制器上, 而对某条期望输出轨迹的跟踪问题的研究很少见。本章将非线性系统的跟踪问题按期望输出的形式分为两种: 对单个稳定期望输出点的逼近问题; 对某条期望输出轨迹的跟踪问题。针对这两种跟踪问题, 首先, 分别设计了非线性预测控制器, 给出了为保证系统稳定所需满足的条件; 其次, 针对第二种跟踪问题给出了一种简单的计算预测控制次优控制律的方法; 最后, 给出了一种基于线性近似的非线性模型预测控制方法, 并在此基础上提出了一种基于 RBF 神经网络逼近的非线性模型预测控制方案。在基于线性近似的预测控制方法中, 首先运用斯特林插值公式法对非线性函数进行线性近似处理, 然后将模型预测控制性能指标重构为一个二次型最优化问题, 通过对二次型最优化问题的求解得到了模型预测控制的最优控制序列。为了降低计算的复杂度, 该 MPC 方案忽略了线性近似过程中产生的非线性高阶项。在基于 RBF 神经网络逼近的非线性模型预测控制方案中, 用 RBF 神经网络逼近线性化过程中产生的非线性高阶项, 提高了模型的精确度。

## 第 6 章

# 预测控制在航天器末端 自主交会中的应用

### 6.1 引言

近年来，很多国家一直在研究航天器的自主交会对接技术，取得了不少成果，在理论上提出了一些很具参考价值的运动模型和控制方法<sup>[115-117]</sup>，俄罗斯也首次实现了两枚航天器的自主交会对接。所谓自主交会对接，指的是航天器不依靠地面站的参与，依赖自身的星载敏感器对两枚航天器（追踪航天器与目标航天器）的相对位置信息、相对速度信息以及相对姿态和相对姿态速率信息进行测量，通过自身的星载计算机输出控制指令，实现两枚航天器的交会对接任务。

本章以航天器的末端交会控制为研究目标，建立两枚航天器的末端交会运动模型，并设计末端交会过程中的控制算法。在末端交会时，两枚航天器相距较近且大致处于同一轨道平面内，为保证精度，追踪航天器一般采用有限推力发动机。为了实现平稳交会，需要对两枚航天器之间的接近速度和相

对姿态速率施加约束，而预测控制处理状态约束和控制约束的能力使得其在航天器末端交会中具有良好的应用前景。本章首先建立了基于视线坐标系的相对运动方程，然后将预测控制应用于航天器的末端交会，在控制过程中采用中心差分粒子滤波器对状态信息进行估计，最后通过仿真验证了算法的可行性。

## 6.2 基于视线坐标系的相对运动方程

基于视线坐标系的自主交会按照视线角的定义分为两种类型：基于惯性坐标系的视线制导交会；基于目标轨道坐标系的视线制导交会。本章采用基于惯性坐标系的视线制导交会，坐标系选取如图 6-1 所示。

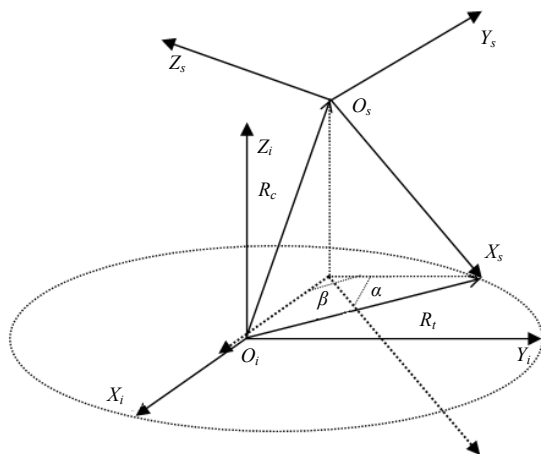


图 6-1 坐标系定义

为方便起见，首先定义相关概念。

惯性坐标系  $X_i Y_i Z_i O_i$ ：原点  $O_i$  为地球质心； $X_i Y_i O_i$  为目标轨道面， $X_i$ 、 $Y_i$  定义在目标轨道面内，按需要选择其具体位置，相互垂直，惯性空间位置

固定，不随地球的自转与目标航天器的运动而运动； $Z_i$  垂直于目标轨道面，与  $X_i$ 、 $Y_i$  构成右手坐标系。

视线坐标系  $X_s Y_s Z_s O_s$ ：在惯性坐标系的基础上，按照  $3\beta - 2\alpha$  转动顺序得到；原点  $O_s$  为追踪航天器质心。

$R_i$ ：目标航天器距离地球质心的径向量。

$R_c$ ：追踪航天器距离地球质心的径向量。

视线  $O_s X_s$ ：追踪航天器指向目标航天器。

视线角  $\beta$ ：视线在  $X_i Y_i O_i$  平面内的投影与  $O_i X_i$  的夹角。

偏离角  $\alpha$ ：视线与  $X_i Y_i O_i$  平面的夹角。

$\mu$ ：地球引力常数。

$r_i$ ：目标航天器与地球质心的距离，即  $r_i = \|R_i\|$ 。

$r_c$ ：追踪航天器与地球质心的距离，即  $r_c = \|R_c\|$ 。

下面以目标航天器与追踪航天器的运动方程为基础，推导基于视线坐标系的相对运动方程<sup>[116,117]</sup>。

目标航天器的运动方程为

$$\ddot{R}_i = -\frac{\mu R_i}{r_i^3} + f_i \quad (6-1)$$

其中， $-\frac{\mu R_i}{r_i^3}$  为目标航天器所受的地球引力； $f_i$  为目标航天器轨道摄动。

追踪航天器的运动方程为

$$\ddot{R}_c = -\frac{\mu R_c}{r_c^3} + f_c + u_c \quad (6-2)$$

其中,  $-\frac{\mu R_c}{r_c^3}$  为追踪航天器所受的地球引力;  $f_c$  为追踪航天器轨道摄动;  $u_c$  为追踪航天器的控制加速度向量。

用式 (6-2) 减去式 (6-1) (因视线方向是由追踪航天器指向目标航天器, 所以应为目标航天器的运动方程减去追踪航天器的运动方程) 可得

$$\left. \frac{d^2 \vec{\rho}}{dt^2} \right|_i = \Delta g + \Delta f - u_c \quad (6-3)$$

其中,  $\vec{\rho} = R_t - R_c$  为相对视线距离向径;  $\left. \frac{d^2 \vec{\rho}}{dt^2} \right|_i = \ddot{R}_t - \ddot{R}_c$  为相对视线距离向径的二次导数;  $\Delta g = -\frac{\mu R_t}{r_t^3} + \frac{\mu R_c}{r_c^3}$  为两枚航天器的引力差;  $\Delta f = f_t - f_c$  为两枚航天器的轨道摄动差。

因为要建立的是基于视线坐标系的相对运动方程, 所以需要将  $\left. \frac{d^2 \vec{\rho}}{dt^2} \right|_i$  投影到视线坐标系上, 经推导可得

$$\left. \frac{d^2 \vec{\rho}}{dt^2} \right|_i = \left. \frac{d^2 \vec{\rho}}{dt^2} \right|_s + 2 \times \omega_{is} \times \left. \frac{d \vec{\rho}}{dt} \right|_s + \dot{\omega}_{is} \times \vec{\rho} + \omega_{is} \times \omega_{is} \times \vec{\rho} \quad (6-4)$$

定义

$$\ddot{\vec{\rho}} = \left. \frac{d^2 \vec{\rho}}{dt^2} \right|_s; \quad \dot{\vec{\rho}} = \left. \frac{d \vec{\rho}}{dt} \right|_s$$

可得

$$\left. \frac{d^2 \vec{\rho}}{dt^2} \right|_i = \ddot{\vec{\rho}} + 2 \times \omega_{is} \times \dot{\vec{\rho}} + \dot{\omega}_{is} \times \vec{\rho} + \omega_{is} \times \omega_{is} \times \vec{\rho} \quad (6-5)$$

考虑到

$$[\omega_{is} \times] = \begin{bmatrix} 0 & -\omega_{isz} & \omega_{isy} \\ \omega_{isz} & 0 & -\omega_{isx} \\ -\omega_{isy} & \omega_{isx} & 0 \end{bmatrix} \quad (6-6)$$



$$[\dot{\omega}_{is} \times] = \begin{bmatrix} 0 & -\dot{\omega}_{isz} & \dot{\omega}_{isy} \\ \dot{\omega}_{isz} & 0 & -\dot{\omega}_{isx} \\ -\dot{\omega}_{isy} & \dot{\omega}_{isx} & 0 \end{bmatrix} \quad (6-7)$$

并且根据视线角与偏离角定义, 可以得到

$$\begin{bmatrix} \omega_{isx} \\ \omega_{isy} \\ \omega_{isz} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \alpha \\ 0 & 1 & 0 \\ 0 & 0 & \cos \alpha \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} -\sin \alpha \dot{\beta} \\ \dot{\alpha} \\ \cos \alpha \dot{\beta} \end{bmatrix} \quad (6-8)$$

$$\begin{bmatrix} \dot{\omega}_{isx} \\ \dot{\omega}_{isy} \\ \dot{\omega}_{isz} \end{bmatrix} = \begin{bmatrix} -\dot{\alpha} \dot{\beta} \cos \alpha - \ddot{\beta} \sin \alpha \\ \ddot{\alpha} \\ -\dot{\alpha} \dot{\beta} \sin \alpha + \ddot{\beta} \cos \alpha \end{bmatrix} \quad (6-9)$$

将式 (6-8) 代入式 (6-6), 将式 (6-9) 代入式 (6-7), 可以得到

$$[\omega_{is} \times] = \begin{bmatrix} 0 & -\dot{\beta} \cos \alpha & \dot{\alpha} \\ \dot{\beta} \cos \alpha & 0 & \dot{\beta} \sin \alpha \\ -\dot{\alpha} & -\dot{\beta} \sin \alpha & 0 \end{bmatrix} \quad (6-10)$$

$$[\dot{\omega}_{is} \times] = \begin{bmatrix} 0 & \dot{\alpha} \dot{\beta} \sin \alpha - \ddot{\beta} \cos \alpha & \ddot{\alpha} \\ -\dot{\alpha} \dot{\beta} \sin \alpha + \ddot{\beta} \cos \alpha & 0 & \dot{\alpha} \dot{\beta} \cos \alpha + \ddot{\beta} \sin \alpha \\ -\ddot{\alpha} & -\dot{\alpha} \dot{\beta} \cos \alpha - \ddot{\beta} \sin \alpha & 0 \end{bmatrix} \quad (6-11)$$

将式 (6-10)、式 (6-11) 代入式 (6-5), 并定义  $\rho$  为相对视线距离,  $\dot{\rho}$  为相对视线距离速率,  $\ddot{\rho}$  为相对视线距离加速度, 则有

$$\vec{\rho} = [\rho \ 0 \ 0]^T$$

$$\dot{\vec{\rho}} = [\dot{\rho} \ 0 \ 0]^T$$

$$\ddot{\vec{\rho}} = [\ddot{\rho} \ 0 \ 0]^T$$

最终可以得到

$$\left. \frac{d^2 \vec{\rho}}{dt^2} \right|_i = \begin{bmatrix} \ddot{\rho} - \left( (\dot{\alpha})^2 + (\dot{\beta} \cos \alpha)^2 \right) \rho \\ \ddot{\beta} \rho \cos \alpha + \dot{\beta} \left( 2\dot{\rho} \cos \alpha - 2\dot{\alpha} \dot{\rho} \sin \alpha \right) \\ -\ddot{\alpha} \rho - 2\dot{\alpha} \dot{\rho} - \sin \alpha \cos \alpha (\dot{\beta})^2 \rho \end{bmatrix} \quad (6-12)$$

将式 (6-12) 写成分量形式:

$$\ddot{\rho} - \left( (\dot{\alpha})^2 + (\dot{\beta} \cos \alpha)^2 \right) \rho = \Delta g_x + \Delta f_x - u_x \quad (6-13)$$

$$\ddot{\beta} \rho \cos \alpha + \dot{\beta} \left( 2\dot{\rho} \cos \alpha - 2\dot{\alpha} \dot{\rho} \sin \alpha \right) = \Delta g_y + \Delta f_y - u_y \quad (6-14)$$

$$-\ddot{\alpha} \rho - 2\dot{\alpha} \dot{\rho} - \sin \alpha \cos \alpha (\dot{\beta})^2 \rho = \Delta g_z + \Delta f_z - u_z \quad (6-15)$$

其中,  $u_x$ 、 $u_y$ 、 $u_z$  为视线坐标系上追踪器的控制加速度;  $\Delta g_x$ 、 $\Delta g_y$ 、 $\Delta g_z$  为视线坐标系上两枚航天器的引力差;  $\Delta f_x$ 、 $\Delta f_y$ 、 $\Delta f_z$  为视线坐标系上两枚航天器的摄动差。

在最终逼近段, 追踪航天器与目标航天器基本处于同一轨道面上, 且距离比较小, 所以偏离角  $\alpha$  可近似看成零; 引力差与摄动差都很小, 将其看成噪声项处理。

因此, 可以得到二维坐标系内的运动方程如下:

$$\ddot{\rho} - (\dot{\beta})^2 \rho = w_x - u_x \quad (6-16)$$

$$\ddot{\beta} \rho + 2\dot{\beta} \dot{\rho} = w_y - u_y \quad (6-17)$$

其中,  $w_x = \Delta g_x + \Delta f_x$ ,  $w_y = \Delta g_y + \Delta f_y$ , 可将其看成噪声项处理。

式 (6-16) 称为追踪航天器在视线坐标系内的纵向运动, 式 (6-17) 称

为追踪航天器在视线坐标系内的横向运动。纵向与横向的运动由单独的发动机控制，分别称为纵向发动机与横向发动机。

为与前面章节对应，将此相对运动方程转变为离散形式。考虑采样时间  $T_s$ ，用一阶龙格库塔法将式 (6-16) 和式 (6-17) 离散化为

$$\rho_{k+1} = \rho_k + T_s \dot{\rho}_k \quad (6-18)$$

$$\dot{\rho}_{k+1} = \dot{\rho}_k + T_s \left( \dot{\beta}_k \right)^2 \rho_k - T_s u_{x,k} + T_s w_{x,k} \quad (6-19)$$

$$\beta_{k+1} = \beta_k + T_s \dot{\beta}_k \quad (6-20)$$

$$\dot{\beta}_{k+1} = \dot{\beta}_k + T_s \left( \frac{-2\dot{\beta}_k \dot{\rho}_k}{\rho_k} \right) - T_s \left( \frac{1}{\rho_k} \right) u_{y,k} + T_s \left( \frac{1}{\rho_k} \right) w_{y,k} \quad (6-21)$$

为方便起见，将式 (6-18)、式 (6-19)、式 (6-20)、式 (6-21) 规整为向量形式：

$$x_{k+1} = f(x_k, u_k, w_k) \quad (6-22)$$

其中， $x_k = (\rho_k, \dot{\rho}_k, \beta_k, \dot{\beta}_k)^T$  为状态向量； $u_k = (u_{x,k}, u_{y,k})^T$  为控制输入； $w_k = (w_{x,k}, w_{y,k})^T$  为过程噪声。

追踪航天器可依赖自身的星载敏感器对两枚航天器的相对视线距离、相对视线距离速率及视线角和视线角速率进行测量，量测方程可描述为

$$y_k = Hx_k + v_k \quad (6-23)$$

其中， $v_k$  为量测噪声；

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

至此，基于视线坐标系的相对运动方程及观测方程就已得出，下一节将通过计算机仿真验证预测控制在末端交会中的应用。

## 6.3 应用仿真

在本次末端交会仿真中，采样时间取  $T_s = 1\text{s}$ ，将其代入式 (6-18) ~ 式 (6-21)，则末端交会的相对运动方程为

$$\begin{cases} \rho_{k+1} = \rho_k + \dot{\rho}_k \\ \dot{\rho}_{k+1} = \dot{\rho}_k + \left(\dot{\beta}_k\right)^2 \rho_k - u_{x,k} + w_{x,k} \\ \beta_{k+1} = \beta_k + \dot{\beta}_k \\ \dot{\beta}_{k+1} = \dot{\beta}_k + \left(\frac{-2\dot{\beta}_k\dot{\rho}_k}{\rho_k}\right) - \left(\frac{1}{\rho_k}\right)u_{y,k} + \left(\frac{1}{\rho_k}\right)w_{y,k} \end{cases}$$

0 时刻的初始状态为

$$\rho_0 = 200\text{m}; \dot{\rho}_0 = 0.5\text{m/s}; \beta_0 = 120^\circ; \dot{\beta}_0 = -0.03^\circ/\text{s}$$

目标状态为

$$\rho_{\text{ref}} = 5\text{m}; \dot{\rho}_{\text{ref}} = 0\text{m/s}; \beta_{\text{ref}} = 180^\circ; \dot{\beta}_{\text{ref}} = 0^\circ/\text{s}$$

过程噪声为

$$w = (w_x, w_y)^T = (\text{rand}[0.01, 0.02], \text{rand}[-0.02, 0])^T$$

其中， $\text{rand}[0.01, 0.02]$  为位于区间  $[0.01, 0.02]$  上的均值噪声； $\text{rand}[-0.02, 0]$  为位于区间  $[-0.02, 0]$  上的均值噪声。

追踪航天器上星载敏感器的量测误差为

$$v = (\mathcal{N}(0, 0.2), \mathcal{N}(0, 0.1), \mathcal{N}(0, 0.01), \mathcal{N}(0, 0.01))^T$$

其中， $\mathcal{N}(0, 0.2)$  表示均值为 0 且方差为 0.2 的高斯白噪声，其他类似。

状态约束为

$$X = \left\{ (\rho, \dot{\rho}, \beta, \dot{\beta}) \left| \begin{array}{l} -2\text{m} \leq \rho - \rho_{\text{ref}} \leq 205\text{m} \\ -1\text{m/s} \leq \dot{\rho} - \dot{\rho}_{\text{ref}} \leq 1\text{m/s} \\ -80^\circ \leq \beta - \beta_{\text{ref}} \leq 80^\circ \\ -5^\circ/\text{s} \leq \dot{\beta} - \dot{\beta}_{\text{ref}} \leq 5^\circ/\text{s} \end{array} \right. \right\}$$

控制约束为

$$U = \left\{ (u_x, u_y) \left| \begin{array}{l} -0.2\text{m/s}^2 \leq u_x \leq 0.2\text{m/s}^2 \\ -0.2\text{m/s}^2 \leq u_y \leq 0.2\text{m/s}^2 \end{array} \right. \right\}$$

本次仿真采用 4.3 节中的鲁棒预测控制算法, 即基于优化线性反馈控制律的鲁棒预测控制。考虑到控制量被约束在很小的幅值范围之内, 所以用饱和控制律来替代线性反馈控制律。取预测步长  $N = 20$ , 考虑到

$$\begin{aligned} x_k &= (\rho_k, \dot{\rho}_k, \beta_k, \dot{\beta}_k)^T; u_k = (u_{x,k}, u_{y,k})^T; \\ w_k &= (w_{x,k}, w_{y,k})^T; x_{\text{ref}} = (\rho_{\text{ref}}, \dot{\rho}_{\text{ref}}, \beta_{\text{ref}}, \dot{\beta}_{\text{ref}})^T \end{aligned}$$

并取阶段指标函数为  $q(x, u) = x^T Q x$ , 其中

$$Q = \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0.1 \end{bmatrix}$$

则预测控制在  $k$  时刻的优化问题为

$$\begin{aligned} & J_N(K, \hat{x}_k, w) \\ &= \min_K \max_w \sum_{i=0}^{N-1} (x(i, \hat{x}_k) - x_{\text{ref}})^T Q (x(i, \hat{x}_k) - x_{\text{ref}}) + \\ & \quad (x(N, \hat{x}_k) - x_{\text{ref}})^T P (x(N, \hat{x}_k) - x_{\text{ref}}) \end{aligned}$$

$$\begin{aligned}
 \text{s.t. } & x(i+1, \hat{x}_k) = f(x(i, \hat{x}_k), \text{Sat}_i(x(i, \hat{x}_k) - x_{\text{ref}})) + w_i \\
 & x(i, \hat{x}_k) \in X, \forall w_i \in W, i = 0, 1, 2, \dots, N-1 \\
 & (x(N, \hat{x}_k) - x_{\text{ref}}) \in \Omega \\
 & \text{Sat}_i(x(i, \hat{x}_k) - x_{\text{ref}}) = (u_x, u_y)^T \\
 u_x = & \begin{cases} 0.2, & K_i(x(i, \hat{x}_k) - x_{\text{ref}})_{(1,1)} > 0.2 \\ K_i(x(i, \hat{x}_k) - x_{\text{ref}})_{(1,1)}, & \|K_i(x(i, \hat{x}_k) - x_{\text{ref}})_{(1,1)}\| \leq 0.2 \\ -0.2, & K_i(x(i, \hat{x}_k) - x_{\text{ref}})_{(1,1)} < -0.2 \end{cases} \\
 u_y = & \begin{cases} 0.2, & K_i(x(i, \hat{x}_k) - x_{\text{ref}})_{(2,1)} > 0.2 \\ K_i(x(i, \hat{x}_k) - x_{\text{ref}})_{(2,1)}, & \|K_i(x(i, \hat{x}_k) - x_{\text{ref}})_{(2,1)}\| \leq 0.2 \\ -0.2, & K_i(x(i, \hat{x}_k) - x_{\text{ref}})_{(2,1)} < -0.2 \end{cases}
 \end{aligned}$$

其中,  $K = \{K_i\}$ ;  $U_{\max} = (0.2, 0.2)^T$ ,  $U_{\min} = (-0.2, -0.2)^T$ ;  $x(N, x_0)^T P x(N, x_0)$  为终端代价函数;  $\hat{x}_k$  为  $k$  时刻状态的估计值。考虑到过程噪声并不满足高斯分布, 且均值并不为零, 我们在对状态进行估计时采用中心差分粒子滤波方法。

拟采用 4.3.3 节中的方法求取终端代价函数中的正定矩阵, 首先将末端交会相对运动方程围绕目标点  $x_{\text{ref}} = (\rho_{\text{ref}}, \dot{\rho}_{\text{ref}}, \beta_{\text{ref}}, \dot{\beta}_{\text{ref}})^T$  进行一阶泰勒展开处理为近似线性系统

$$x_{k+1} = A x_k + B u_k + C w_k$$

其中

$$A = \begin{bmatrix} 1 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix}; B = \begin{bmatrix} 0 & 0 \\ -T_s & 0 \\ 0 & 0 \\ 0 & -0.2T_s \end{bmatrix}; C = \begin{bmatrix} 0 & 0 \\ T_s & 0 \\ 0 & 0 \\ 0 & 0.2T_s \end{bmatrix}$$

其次, 求取使得  $A_{\text{KR}} = A + B K_{\text{loc}}$  的特征值都处于单位圆内的线性反馈控制增益。

$$K_{\text{loc}} = \begin{bmatrix} 0.618 & 1.618 & 0 & 0 \\ 0 & 0 & 4.580 & 9.580 \end{bmatrix}$$

最后，设置可调参数  $\gamma = 0.8$ ，求取非负定矩阵  $\mathbf{P}$ ，使其满足

$$\mathbf{A}_{\text{KR}}^T \mathbf{P} \mathbf{A}_{\text{KR}} - \mathbf{P} + \mathbf{Q} < 0$$

$$\mathbf{B}^T \mathbf{P} \mathbf{B} - \gamma \mathbf{I} < 0$$

得到的结果为

$$\mathbf{P} = \begin{bmatrix} 0.393 & 0.243 & 0 & 0 \\ 0.243 & 0.393 & 0 & 0 \\ 0 & 0 & 31.374 & 16.374 \\ 0 & 0 & 16.374 & 17.874 \end{bmatrix}$$

考虑将饱和控制的稳定域用作预测控制的终端状态约束集，经计算得到的终端状态约束集可取为

$$\Omega = \left\{ x \in X \mid (x - x_{\text{ref}})^T \mathbf{P} (x - x_{\text{ref}}) \leq 1.509 \times 10^4 \right\}$$

将  $\mathbf{P}$  和  $\Omega$  带入预测控制的优化问题，并采用滚动控制策略，经过仿真，得到如图 6-2~图 6-7 所示的结果。

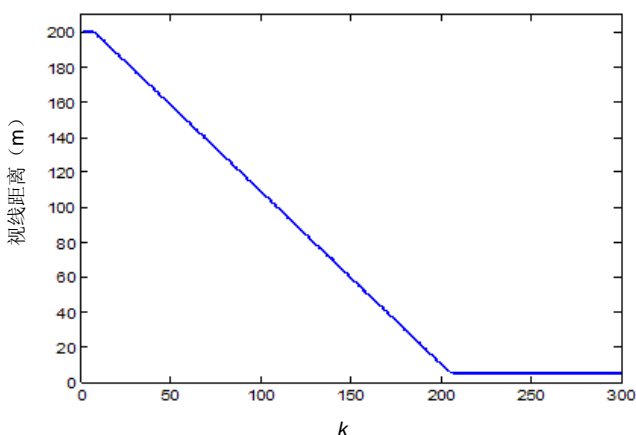


图 6-2 视线距离变化曲线

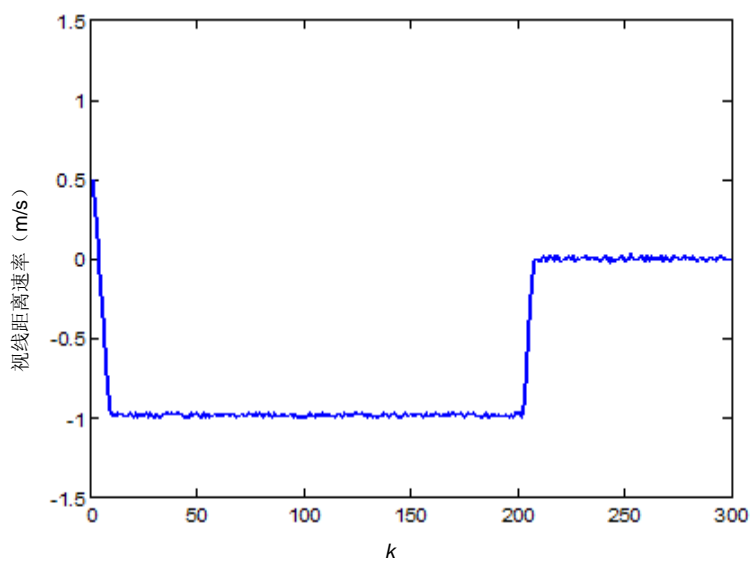


图 6-3 视线距离速率变化曲线

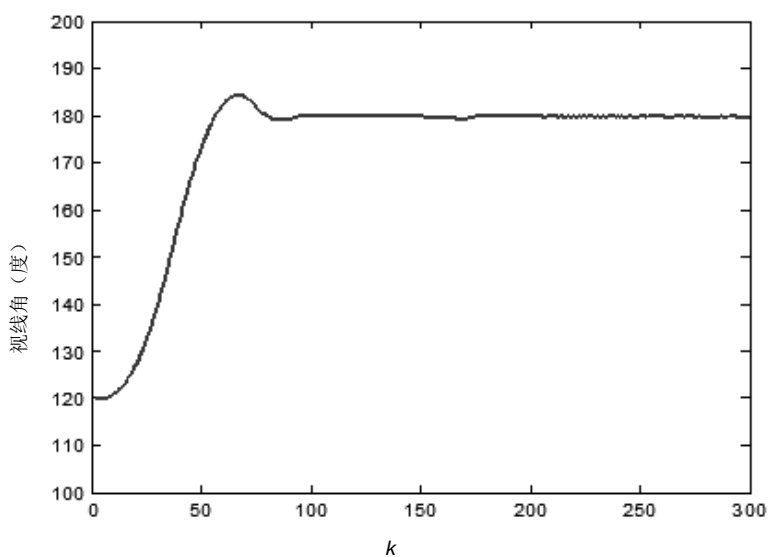


图 6-4 视线角变化曲线



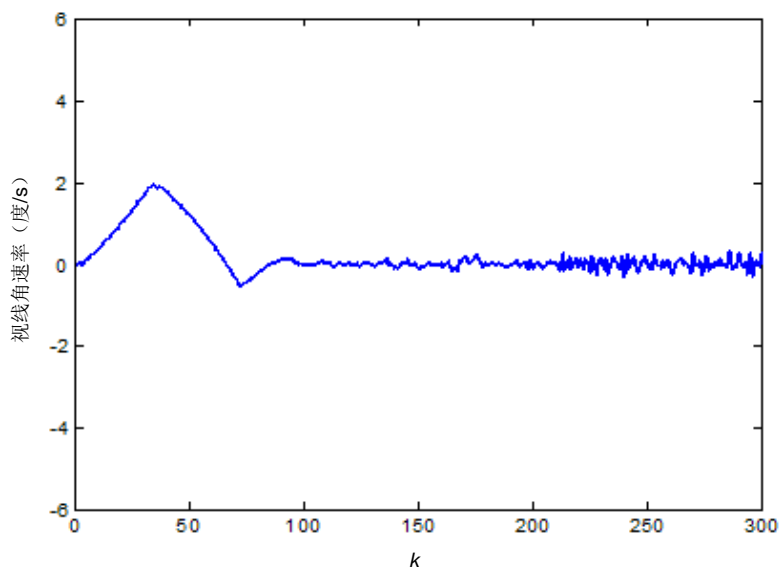


图 6-5 视线角速率变化曲线

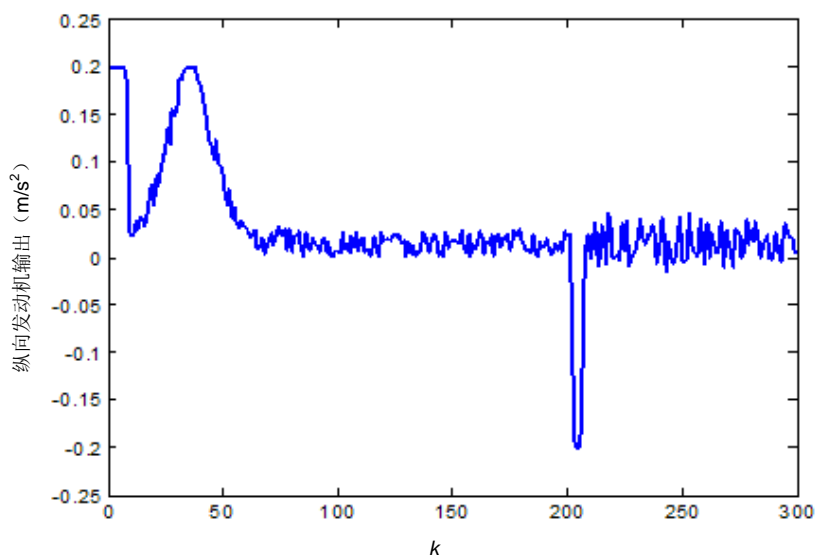


图 6-6 纵向控制输出

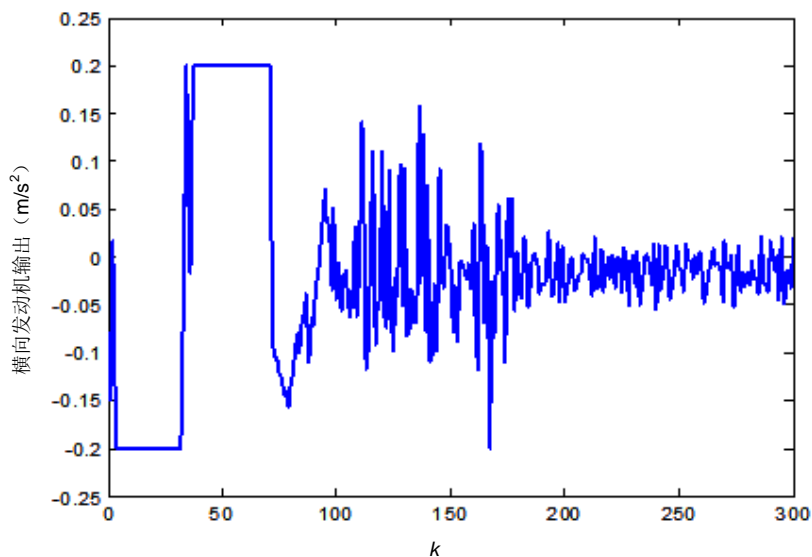


图 6-7 横向控制输出

由图 6-2~图 6-7 可以看出, 视线距离、视线角都可被控制至目标点, 而且视线距离速率、视线角速率都满足约束条件; 纵向控制指令和横向控制指令也都满足纵向发动机与横向发动机的输出约束。

## 6.4 本章小结

关于航天器的末端交会控制, 现有方法通常采用滑模变结构控制。而滑模变结构控制在处理一些约束条件 (如控制量幅值约束、状态量幅值约束) 时显得比较困难。考虑到预测控制处理约束条件的能力, 本章将预测控制方法应用于航天器的末端交会控制, 并采用中心差分粒子滤波器对两枚航天器的相对视线距离、视线距离速率、视线角、视线角速率进行估计, 通过计算机仿真验证了预测控制方法应用于末端交会控制的可行性。

## 第 7 章

# 模型预测控制在飞行器 姿态机动中的应用

### 7.1 引言

在现代空间任务中，如航天器空间交会对接、空天飞行器再入段姿态调整、飞行器的姿态捕获、飞行器空间光通信、近地小卫星的编队飞行等，通常要求飞行器具有良好的姿态机动能力<sup>[137]</sup>。在无人飞行器，尤其是大型无人机自动着陆或着舰的过程中，需要飞行器以预设的姿态降落，以使飞行器机体平面与跑道平面啮合，达到最好的着陆或着舰效果，这实际上也是一类飞行器姿态机动问题。描述飞行器姿态的系统通常是具有强耦合特性的多输入多输出（MIMO）非线性系统，这为解决飞行器姿态控制问题带来了巨大困难。传统控制方案一般先通过特征点局部线性化及系数“冻结”法来获得飞行器姿态模型，然后按线性系统的控制方案进行设计<sup>[138]</sup>，这要求原始模型必须相对精确，而对姿态进行调整的范围却十分有限。滑模变结构控制方法是能较好地处理非线性问题的方法之一<sup>[139]</sup>，但当对高阶（MIMO）非线性

系统进行处理时，往往需要在多个滑模面上进行切换，这大大增加了计算的复杂度。并且由于切换开关的存在，还会给系统带来高频“抖振”，从而影响系统的性能指标。模型预测控制（Model Predictive Control, MPC）是一种基于滚动优化的在线控制策略，具有对模型要求低、抗干扰性好、鲁棒性强等优点，而且能够在优化性能指标的同时较好地处理各种约束条件，因此近年来得到了众多工程技术人员和理论研究者的重视，并被应用到飞行器姿态控制问题的研究中<sup>[140-144]</sup>。

本章针对飞行器的姿态机动问题，提出了一种将飞行器姿态驱动到设定姿态的模型预测控制算法。由于模型预测控制的一个重要特征是能实现对设定参考点或参考轨迹的有效跟踪，因此本章要解决的飞行器姿态机动问题可以转换为对设定姿态的跟踪问题。即将设定的机动姿态作为参考点，如果能够通过对某一性能指标的滚动优化，在某一时域内使飞行器的姿态成功跟踪参考点姿态，那么就实现了飞行器的姿态机动。如果飞行器的设定姿态多于一个，则可转换为对多个参考点组成的参考轨迹的跟踪问题。仿真结果表明，本章的模型预测控制算法成功实现了飞行器的姿态机动。

## 7.2 飞行器姿态运动方程的转换与解耦

本章考虑的飞行器姿态运动方程是基于机体坐标轴系描述的，如图 7-1 所示。原点  $O$  为飞行器的质心， $x$  轴垂直于飞机纵向对称平面指向飞行器的两侧，从机尾看向右为正； $y$  轴在飞机对称面内并平行于飞机的纵轴线，机头方向为正； $z$  轴由  $x$  轴和  $y$  轴按右手螺旋定则确定，一般向上为正。

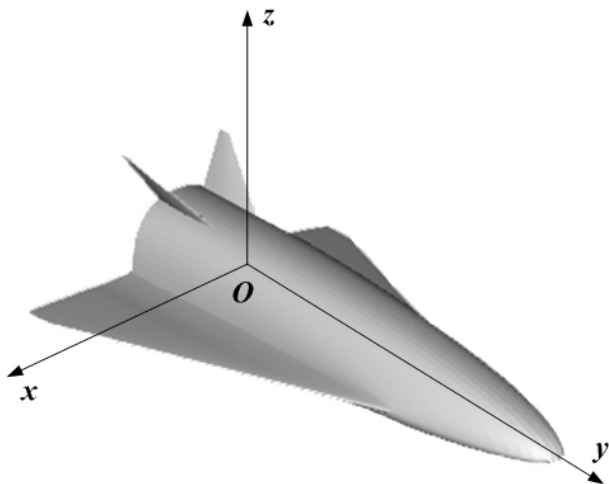


图 7-1 坐标系定义

在建立飞行器姿态运动方程之前，首先做出如下假设<sup>[145,146]</sup>：

- (1) 飞行器为刚体且质量保持不变。
- (2) 地球曲率为零，即将地球表面看作平面。
- (3) 重力加速度为常值，不随飞行高度变化。

(4) 飞行器为面对称的，对称平面为机体坐标轴系的  $Oyz$  平面，飞行器的几何外形和内部质量都是对称分布的。

设  $\theta_1$ 、 $\theta_2$ 、 $\theta_3$  分别为本章所研究刚体飞行器的滚转角、偏航角和俯仰角； $\omega_1$ 、 $\omega_2$ 、 $\omega_3$  分别为飞行器绕  $y$  轴、 $z$  轴和  $x$  轴的旋转角速率； $J_1$ 、 $J_2$ 、 $J_3$  分别为飞行器关于  $y$  轴、 $z$  轴和  $x$  轴的转动惯量； $L_1$ 、 $L_2$ 、 $L_3$  分别为飞行器关于  $y$  轴、 $z$  轴和  $x$  轴所受到的控制力矩。则飞行器姿态运动方程可描述为<sup>[147]</sup>

$$\begin{cases} \dot{\theta}_1 = \omega_1 - \omega_2 \cos \theta_1 \cdot \tan \theta_3 + \omega_3 \sin \theta_1 \cdot \tan \theta_3 \\ \dot{\theta}_2 = \frac{\cos \theta_1}{\cos \theta_3} \omega_2 - \frac{\sin \theta_1}{\cos \theta_3} \omega_3 \\ \dot{\theta}_3 = \omega_3 \cos \theta_1 + \omega_2 \sin \theta_1 \\ \dot{\omega}_1 = \frac{J_2 - J_3}{J_1} \omega_2 \omega_3 + \frac{L_1}{J_1} \\ \dot{\omega}_2 = \frac{J_3 - J_1}{J_2} \omega_3 \omega_1 + \frac{L_2}{J_2} \\ \dot{\omega}_3 = \frac{J_1 - J_2}{J_3} \omega_1 \omega_2 + \frac{L_3}{J_3} \end{cases} \quad (7-1)$$

由式(7-1)可知飞行器姿态运动方程为非线性的,且各通道之间存在较强的耦合作用。本章按文献[147]的方法对飞行器姿态运动方程进行反馈线性化及通道间解耦。

首先给出式(7-1)的状态空间描述。作如下定义:

$$\begin{cases} x_1 = \theta_1 \\ x_2 = \theta_2 \\ x_3 = \theta_3 \\ x_4 = \omega_1 \\ x_5 = \omega_2 \\ x_6 = \omega_3 \end{cases}, \quad \begin{cases} \lambda_1 = \frac{J_2 - J_3}{J_1} \\ \lambda_2 = \frac{J_3 - J_1}{J_2} \\ \lambda_3 = \frac{J_1 - J_2}{J_3} \end{cases}, \quad \begin{cases} u_1 = \frac{L_1}{J_1} \\ u_2 = \frac{L_2}{J_2} \\ u_3 = \frac{L_3}{J_3} \end{cases}$$

取飞行器的输出为三个姿态角,则可用状态空间方程对飞行器姿态运动方程进行描述,具体形式为

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}(\mathbf{x}) + \mathbf{B}\mathbf{u} \\ \mathbf{y} = \mathbf{C}\mathbf{x} \end{cases} \quad (7-2)$$

其中

$$\begin{aligned} \mathbf{x} &= (x_1, x_2, x_3, x_4, x_5, x_6)^T \\ \mathbf{u} &= (u_1, u_2, u_3)^T \end{aligned}$$

$$\mathbf{y} = (y_1, y_2, y_3)^T$$

$$\mathbf{A}(\mathbf{x}) = \begin{pmatrix} x_4 - x_5 \cos x_1 \cdot \tan x_3 + x_6 \sin x_1 \cdot \tan x_3 \\ \frac{\cos x_1}{\cos x_3} x_5 - \frac{\sin x_1}{\cos x_3} x_6 \\ x_6 \cos x_1 + x_5 \sin x_1 \\ \lambda_1 x_5 x_6 \\ \lambda_2 x_6 x_4 \\ \lambda_3 x_4 x_5 \end{pmatrix}$$

$$\mathbf{B} = (\mathbf{0}_{3 \times 3}, \mathbf{I}_{3 \times 3})^T$$

$$\mathbf{C} = (\mathbf{I}_{3 \times 3}, \mathbf{0}_{3 \times 3})$$

接下来用反馈线性化方法实现对姿态运动方程的线性化处理和各通道间解耦。作如下状态变换：

$$\begin{cases} z_1^0 = x_1 \\ z_1^1 = x_4 - x_5 \cos x_1 \cdot \tan x_3 + x_6 \sin x_1 \cdot \tan x_3 \\ z_2^0 = x_2 \\ z_2^1 = \frac{\cos x_1}{\cos x_3} x_5 - \frac{\sin x_1}{\cos x_3} x_6 \\ z_3^0 = x_3 \\ z_3^1 = x_6 \cos x_1 + x_5 \sin x_1 \end{cases}$$

为了确定上述状态变换的可逆性，写出  $\mathbf{z} = \mathbf{z}(\mathbf{x})$  的雅可比矩阵为

$$\nabla \mathbf{z} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ (x_5 \sin x_1 + x_6 \cos x_1) \tan x_3 & 0 & (x_6 \sin x_1 - x_5 \cos x_1) \sec^2 x_3 & 1 & -\cos x_1 \cdot \tan x_3 & \sin x_1 \cdot \tan x_3 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -(x_5 \sin x_1 + x_6 \cos x_1) \sec x_3 & 0 & (x_5 \cos x_1 - x_6 \sin x_1) \sec x_3 \cdot \tan x_3 & 0 & \cos x_1 \cdot \sec x_3 & -\sin x_1 \cdot \sec x_3 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ x_5 \cos x_1 - x_6 \sin x_1 & 0 & 0 & 0 & \sin x_1 & \cos x_1 \end{bmatrix}$$

其中， $\mathbf{z} = (z_1^0, z_1^1, z_2^0, z_2^1, z_3^0, z_3^1)^T$ 。计算  $\nabla \mathbf{z}$  的行列式  $\det(\nabla \mathbf{z}) = -\sec x_3$  可知其值恒不为 0，而  $\mathbf{z} = \mathbf{z}(\mathbf{x})$  显然是光滑的，因此由反函数存在定理可知，非线性

状态变换  $z = z(x)$  为全局有意义的。

对控制输入进行变换如下：

$$v = P + Hu \quad (7-3)$$

其中

$$v = (v_1, v_2, v_3)^T$$

$$P = (P_1, P_2, P_3)^T$$

$$H = (H_1^T, H_2^T, H_3^T)^T$$

$$P_1 = (x_5 \sin x_1 + x_6 \cos x_1) \left[ (x_4 + x_6 \sin x_1 \cdot \tan x_3 - x_5 \cos x_1 \cdot \tan x_3) \tan x_3 + \right. \\ \left. (x_6 \sin x_1 - x_5 \cos x_1) \sec^2 x_3 \right] + \lambda_1 x_5 x_6 - \lambda_2 x_4 x_6 \cos x_1 \cdot \tan x_3 + \lambda_3 x_4 x_5 \sin x_1 \cdot \tan x_3$$

$$P_2 = (x_5 \sin x_1 + x_6 \cos x_1) (2x_5 \cos x_1 \cdot \tan x_3 - 2x_6 \sin x_1 \cdot \tan x_3 - x_4) \sec x_3 + \\ x_4 (\lambda_2 x_6 \cos x_1 - \lambda_3 x_5 \sin x_1) \sec x_3$$

$$P_3 = (x_4 + x_6 \sin x_1 \cdot \tan x_3 - x_5 \cos x_1 \cdot \tan x_3) (x_5 \cos x_1 - x_6 \sin x_1) + \\ x_4 (\lambda_2 x_6 \sin x_1 + \lambda_3 x_5 \cos x_1)$$

$$H_1 = (1, \cos x_1 \cdot \tan x_3, \sin x_1 \cdot \tan x_3)$$

$$H_2 = (0, \cos x_1 \cdot \sec x_3, -\sin x_1 \cdot \sec x_3)$$

$$H_3 = (0, \sin x_1, \cos x_1)$$

通过上述变换过程，可得飞行器姿态控制系统的状态和输出方程为

$$\begin{cases} \dot{z} = A_0 z + B_0 v \\ y = C_0 z \end{cases} \quad (7-4)$$

其中

$$A_0 = \begin{bmatrix} A_0^1 & & \\ & A_0^2 & \\ & & A_0^3 \end{bmatrix}, \quad A_0^i = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad i = 1, 2, 3$$



$$B_0 = \begin{bmatrix} B_0^1 & & \\ & B_0^2 & \\ & & B_0^3 \end{bmatrix}, B_0^i = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, i=1,2,3$$

$$C_0 = \begin{bmatrix} C_0^1 & & \\ & C_0^2 & \\ & & C_0^3 \end{bmatrix}, C_0^i = [1 \ 0], i=1,2,3$$

式(7-4)即经过反馈线性化得到的飞行器姿态控制系统的状态空间描述。由式(7-4)可直接看出,系统式(7-1)已经被解耦成三个相互独立的控制通道,每个通道都为临界稳定的线性能控规范型。

### 7.3 模型预测控制器设计

在7.2节中,我们将飞行器的姿态控制系统式(7-1)进行了线性化解耦,得到了式(7-4)所示的状态空间描述。本节采用连续系统的模型预测控制理论对系统式(7-4)设计模型预测控制器。

首先考虑一般形式的用状态空间描述的连续系统

$$\begin{cases} \dot{x} = A_m x + B_m u \\ y = C_m x \end{cases} \quad (7-5)$$

其中,  $x \in R^n$ ,  $u \in R^m$  和  $y \in R^l$  分别为系统的状态、控制输入和输出。针对上述系统的模型预测控制性能指标函数通常选为如下形式:

$$J = \int_0^{T_p} \left( x(t_i + t' | t_i)^T Q x(t_i + t' | t_i) + \dot{u}(t')^T R \dot{u}(t') \right) dt' \quad (7-6)$$

其中,  $T_p$  为预测步长,  $Q$  和  $R$  为给定的加权矩阵。系统的当前状态为  $x(t_i)$ 。通过最小化代价函数式(7-6)可得滚动优化的最优控制律  $\dot{u}(t')$ 。但是这种对代价函数的描述方式并不能体现系统对设定点或设定轨迹的跟踪效果。

为了在优化问题中考虑到系统对设定点的跟踪，可以选取如下的性能指标函数：

$$J = \int_0^{T_p} \left( (r(t_i) - y(t_i + t' | t_i))^T (r(t_i) - y(t_i + t' | t_i)) + \dot{u}(t')^T \mathbf{R} \dot{u}(t') \right) dt' \quad (7-7)$$

其中， $r(t_i)$  为设定点，在优化时域内为常值（或常值的集合）， $y(t_i + t' | t_i)$  为  $t_i$  时刻预测的  $t_i + t'$  时刻的系统输出，通过对代价函数式（7-7）的最小化，则可以得到驱动系统输出无限接近设定点的控制律。为了将式（7-7）转换为与标准式（7-6）相同的形式，给出如下的增广模型：

$$\begin{bmatrix} \dot{\bar{x}}(t_i + t' | t_i) \\ \dot{e}(t_i + t' | t_i) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_m & \mathbf{0}_m^T \\ \mathbf{C}_m & \mathbf{0}_{l \times l} \end{bmatrix} \begin{bmatrix} \bar{x}(t_i + t' | t_i) \\ e(t_i + t' | t_i) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_m \\ \mathbf{0}_{l \times m} \end{bmatrix} \dot{u}(t') \quad (7-8)$$

其中， $0 \leq t' \leq T_p$ ， $e(t_i + t' | t_i)$  为系统跟踪误差，由式（7-8）可得其值为

$$e(t_i + t' | t_i) = y(t_i + t' | t_i) - r(t_i) = \mathbf{C}_m \bar{x}(t_i + t' | t_i) \quad (7-9)$$

$r(t_i)$  为常值向量，定义如下：

$$r(t_i) = [r_1(t_i) \quad r_2(t_i) \quad \cdots \quad r_l(t_i)]^T$$

将式（7-9）代入式（7-7）中可得

$$\begin{aligned} J &= \int_0^{T_p} \left( (r(t_i) - y(t_i + t' | t_i))^T (r(t_i) - y(t_i + t' | t_i)) + \dot{u}(t')^T \mathbf{R} \dot{u}(t') \right) dt' \\ &= \int_0^{T_p} \left( (-e(t_i + t' | t_i))^T (-e(t_i + t' | t_i)) + \dot{u}(t')^T \mathbf{R} \dot{u}(t') \right) dt' \\ &= \int_0^{T_p} \left( (\mathbf{C}_m \bar{x}(t_i + t' | t_i))^T (\mathbf{C}_m \bar{x}(t_i + t' | t_i)) + \dot{u}(t')^T \mathbf{R} \dot{u}(t') \right) dt' \\ &= \int_0^{T_p} \left( \bar{x}(t_i + t' | t_i)^T \mathbf{C}_m^T \mathbf{C}_m \bar{x}(t_i + t' | t_i) + \dot{u}(t')^T \mathbf{R} \dot{u}(t') \right) dt' \end{aligned}$$

定义  $\mathbf{Q} = \mathbf{C}_m^T \mathbf{C}_m$ ，代入上式中，则式（7-7）可描述为

$$J = \int_0^{T_p} \left( \bar{x}(t_i + t' | t_i)^T \mathbf{Q} \bar{x}(t_i + t' | t_i) + \dot{u}(t')^T \mathbf{R} \dot{u}(t') \right) dt' \quad (7-10)$$

其中,  $\bar{x}(t_i + t'|t_i)$  可由式 (7-8) 得出。针对性能指标函数式 (7-10) 的模型预测控制优化问题可描述为

$$\begin{aligned} & \min_{u \in \mathcal{L}^\infty([0, T_p], R^m)} J(\bar{x}, u) \\ &= \min_{u \in \mathcal{L}^\infty([0, T_p], R^m)} \int_0^{T_p} \left( \bar{x}(t_i + t'|t_i)^T Q \bar{x}(t_i + t'|t_i) + \dot{u}(t')^T R \dot{u}(t') \right) dt' \quad (7-11) \\ &= \min_{u \in \mathcal{L}^\infty([0, T_p], R^m)} \int_0^{T_p} F(\bar{x}(0; t_i), u(t')) dt' \end{aligned}$$

其中,  $F(\bar{x}(0; t_i), u(t')) = \bar{x}(t_i + t'|t_i)^T Q \bar{x}(t_i + t'|t_i) + \dot{u}(t')^T R \dot{u}(t')$  为阶段代价函数,  $\mathcal{L}^\infty(D, R^n)$  为所有可测量且本质有界函数  $\varphi: R \supset D \rightarrow R^n$  的集合。则通过对优化问题式 (7-11) 求解即可得到模型预测控制的最优控制序列

$$u_{\text{MPC}}(t) = u_T^*(t; x_0, 0), 0 \leq t < T_p \quad (7-12)$$

将对象系统式 (7-5) 替换为 7.3 节中给出的飞行器姿态控制系统的状态和输出方程式 (7-4), 重复上述变换过程, 可得  $t_i$  时刻的飞行器姿态跟踪的模型预测控制优化问题如下:

$$\begin{aligned} & \min_{\bar{v} \in \mathcal{L}^\infty([t_i, t_i + T], R^m)} J_T(z_{t_i}, \bar{v}) \\ &= \min_{\bar{v} \in \mathcal{L}^\infty([t_i, t_i + T], R^m)} \int_{t_i}^{t_i + T} F(\bar{z}(t'; t_i), \bar{v}(t')) dt' \\ & \text{s.t.} \quad (7-13) \\ & \dot{\bar{z}}(t'; t_i) = f(\bar{z}(t'; t_i), \bar{v}(t')) \\ & \bar{z}(t'; t_i) \in Z \\ & \bar{v}(t') \in U \end{aligned}$$

其中,  $T$  为预测域,  $t' \in [0, T]$ ,  $Z$  和  $U$  分别为系统的状态约束和输入约束。另外, 对  $t' \in [t_i, t_i + T]$ , 开始于初始状态  $z_{t_i}$  受  $\bar{v}(t')$  驱动的状态预测轨迹用  $\bar{z}(t'; t_i)$  表示。假设使  $J_T(z(t_i), \bar{v})$  取最小值的最优控制量是由  $v_T^*(t'; z_{t_i}, t_i)$ ,  $t' \in [t_i, t_i + T]$  给出的。相应的最优性能指标由  $J_T^*(z_{t_i})$  定义, 相应的预测轨迹为  $z_T^*(t'; z_{t_i}, t_i)$ ,  $t' \in [t_i, t_i + T]$ 。对给定的采样时间  $\delta > 0$ , 系统的控制输入由预测控制优化问题式 (7-13) 在采样时刻  $t_i = i\delta$  的最优解来定义, 为如下形式:

$$v_{\text{MPC}}(t) = v_T^*(t; z_{t_i}, t_i), t_i \leq t < t_i + \delta \quad (7-14)$$

由于本章的模型预测控制算法是在有限时域内的局部优化问题，在对设定姿态进行跟踪时无须将其近似为无穷时域的全局最优化问题，因此未添加终端代价函数约束也能保证系统的闭环稳定性。同时，由于模型预测控制本身的鲁棒性，在滚动优化过程中已实时考虑了外部干扰对飞行器姿态的影响。当给定参考点姿态后，通过对当前飞行器系统状态  $z_0$  的测量及对预测域  $T_p$  和控制域  $T_c$  的设定，在预测步长  $T_N$  内，用本节的模型预测控制算法可实现对设定姿态的跟踪，从而完成飞行器的姿态机动。我们将在 7.4 节用仿真实例对本节针对飞行器姿态机动的模型预测控制算法的有效性进行验证。

## 7.4 应用仿真分析

本节用实例验证 7.3 节中飞行器姿态机动模型预测控制算法的有效性。

作如下假设：某飞行器的姿态运动方程可用式 (7-1) 描述，并可通过反馈线性化转换为式 (7-4) 的状态方程描述。该飞行器执行自动着陆任务，到达自动着陆窗口后，根据当前自身速度及与着陆点的距离计算着陆前时间，作为预测步长  $T_N$ ；根据预测步长设定预测域  $T_p$  和控制域  $T_c$ ；将当前自身参数  $\theta_{10}, \theta_{20}, \theta_{30}, \omega_{10}, \omega_{20}, \omega_{30}, L_{10}, L_{20}, L_{30}$  转换为预测控制初始状态  $z_0$ ；将着陆点时的设定姿态  $r_\theta = (\theta_{1r}, \theta_{2r}, \theta_{3r})$ ,  $r_\omega = (\omega_{1r}, \omega_{2r}, \omega_{3r})$  作为参考点，进行在线滚动优化控制，达到成功跟踪设定着陆姿态的控制目标。为了实现飞行器的平稳安全着陆，需要对飞行器的旋转角速率等进行限制，作为预测控制的约束条件。初始参数设置如下：

预测步长  $T_N = 10 \text{ s}$ ，预测域  $T_p = 4 \text{ s}$ ，控制域  $T_c = 2 \text{ s}$ ；初始姿态角  $(\theta_{10}, \theta_{20}, \theta_{30}) = (-0.5, 0.3, -0.4)$ ，单位为  $\text{rad}$ ；初始角速率  $(\omega_{10}, \omega_{20}, \omega_{30}) = (0, 0, 0)$ ，单位为  $\text{rad/s}$ ；初始控制力矩  $(L_{10}, L_{20}, L_{30}) = (3, -1, 2)$ ，单位为  $\text{N} \cdot \text{m}$ 。

为了使飞行器平稳着陆,参考点姿态设置为

$$\begin{aligned} r_\theta &= (\theta_{1r}, \theta_{2r}, \theta_{3r}) = (0, 0, 0) \\ r_\omega &= (\omega_{1r}, \omega_{2r}, \omega_{3r}) = (0, 0, 0) \\ r_L &= (L_{1r}, L_{2r}, L_{3r}) = (0, 0, 0) \end{aligned}$$

对控制过程中的角速率约束为  $|\omega_i| \leq 0.3 \text{ rad/s}, i=1, 2, 3$ 。

由 7.2 节中的参数设定可得

$$\mathbf{z}_0 = \begin{bmatrix} z_1^0 \\ z_1^1 \\ z_2^0 \\ z_2^1 \\ z_3^0 \\ z_3^1 \end{bmatrix} = \begin{bmatrix} -0.5 \\ 0 \\ 0.3 \\ 0 \\ -0.4 \\ 0 \end{bmatrix}, \mathbf{v} = \mathbf{P} + \mathbf{H}\mathbf{u} = \begin{bmatrix} 1 & -0.37 & 0.2 \\ 0 & 0.95 & 0.52 \\ 0 & -0.48 & 0.88 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

对系统模型式 (7-4) 采用 7.3 节的模型预测控制算法, 通过求解优化问题式 (7-13) 可得到使飞行器对设定参考点姿态进行有效跟踪的最优控制律, 从而实现飞行器的姿态机动。具体仿真结果如图 7-2~图 7-4 所示。其中, 图 7-2 为飞行器姿态角的跟踪效果图, 图 7-3 为飞行器绕各坐标轴的旋转角速率变化图, 图 7-4 为飞行器关于各坐标轴所受到的控制力矩的变化图。由曲线可以看出, 在给定的预测步长内, 模型预测控制器对各姿态角都实现了成功跟踪, 且跟踪过程的角速率均能满足给定的约束。因此, 本章的模型预测控制算法能使飞行器成功实现姿态机动。

本例的方法可应用到对舰载无人机的自动着舰控制中, 但需要对着舰甲板的运动姿态进行准确预报, 并实现舰机间的实时通信。舰载无人机进入着舰窗口后, 以同样的方法计算出着舰前时间。以当前预报的着舰时刻甲板姿态作为参考点, 进行在线滚动优化控制。随着着舰前时间的缩短, 对着舰时刻甲板平面姿态的预报结果会发生变化, 即设定参考点会实时更新。通过舰机间的实时通信, 由对单一参考点的姿态跟踪问题演变为对一个动态参

考点的跟踪问题。因此，对数据精度和技术水平要求更高，预测控制问题也更复杂。

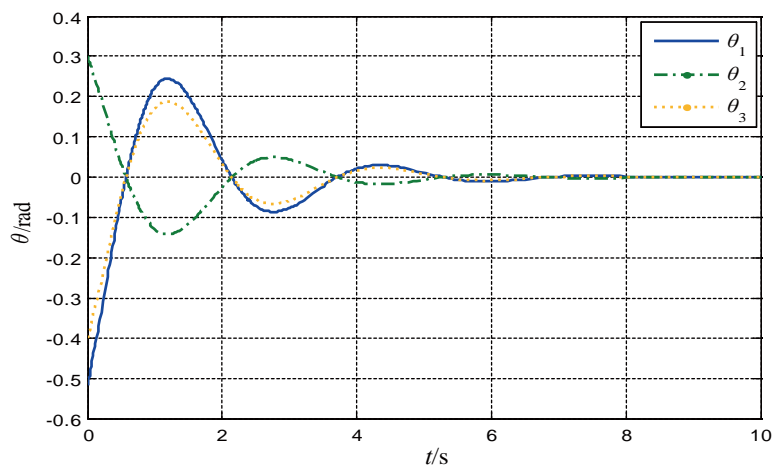


图 7-2 姿态角跟踪曲线

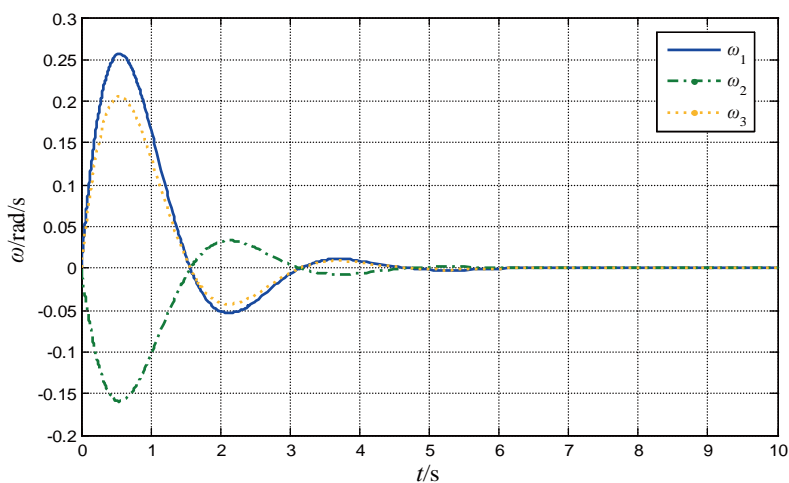


图 7-3 角速率变化曲线

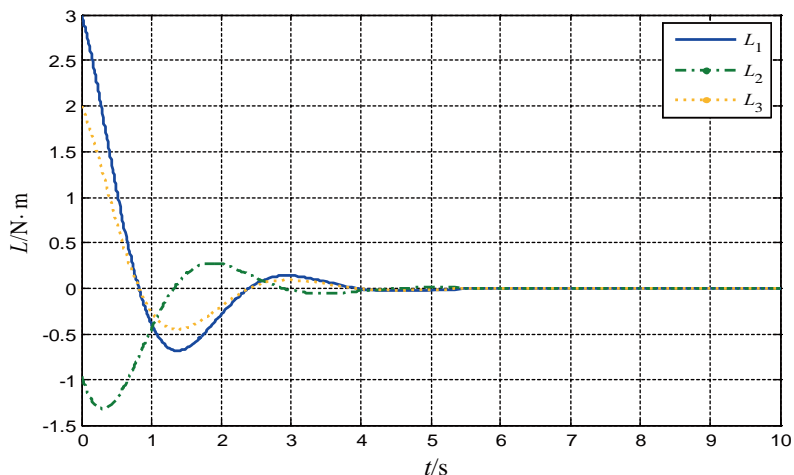


图 7-4 控制力矩变化曲线

如果本例中的飞行器不是执行自动着陆任务，而是在飞行阶段的姿态机动，则对当前参考点姿态进行成功跟踪后，将跟踪后的姿态作为初始状态。重复本例的过程，即可进行对下一个设定参考点姿态的跟踪，从而实现飞行器的空中连续机动。

## 7.5 本章小结

本章针对飞行器的姿态机动问题，提出了一种将飞行器姿态驱动到设定姿态的模型预测控制算法。基于经反馈线性化处理并实现了三通道解耦的飞行器姿态运动系统设计了模型预测控制器。仿真结果表明，本章的模型预测控制算法成功实现了飞行器的姿态机动。在后续研究中，可将本章的方法扩展到舰载机自动着舰和飞行器空中连续机动的姿态控制问题中。

## 第 8 章

# 结论与展望

### 8.1 主要工作总结

预测控制由于其处理约束条件的能力在工业控制中获得了巨大的成功，而针对线性系统的预测控制研究已经相对成熟。但是，实际工业过程往往都存在固有的非线性环节，采用线性模型来描述系统会产生一定偏差，造成控制精度不高，有时甚至会造成系统不稳定。因此，针对非线性系统的预测控制研究就成为控制工程界的重要研究领域。本书总结了前人的研究工作，在此基础上，对非线性预测控制的一些问题进行了较为深入的研究，主要研究成果总结如下：

(1) 针对非线性系统的预测控制，分析了为保证系统稳定，终端状态约束集所需满足的条件，并提出了两种求取终端状态约束集的方法。第一种方法是采用外包集序列逼近预测控制的最终终端状态约束集，从理论上证明了，当迭代次数趋近于无穷步时，外包集序列收敛于最终终端状态约束集。此序列中的每一个外包集均采用支持向量机依次从状态空间中分离得到。第二种方法是采用子集序列逼近预测控制的最终终端状态约束集，同样，从理



论上证明了在迭代次数趋近于无穷时,子集序列收敛于最大终端状态约束集。而此序列中的每一个子集也采用支持向量机依次从状态空间中分离得到。

(2) 分析了非线性预测控制的终端代价函数分别与终端状态约束集和预测控制消耗的全局性能指标之间的关系。证明了按比例增大终端代价函数,预测控制的终端状态约束集会增大,但同时采用预测控制所消耗的全局性能指标也会增多。给出了一种简单的求取终端代价函数的方法:首先,将非线性系统在原点处线性化展开处理为近似线性系统;其次,利用线性最优控制理论得到一个初始的终端代价函数;最后,以终端代价函数与终端状态约束集及终端代价函数与全局性能指标之间的定性关系为准则,按照我们的需要对已经得到的终端代价函数进行比例调整得到最终的终端代价函数。

(3) 针对存在有界干扰时的非线性系统,设计了三种鲁棒预测控制算法。第一种为以集合包含度为优化指标的鲁棒预测控制算法,该方法以终端状态约束集对近似预测可达集的包含度为优化指标,采用预测步长逐渐减小的控制策略,将系统状态控制至一个鲁棒控制不变集。第二种为基于优化线性反馈控制律的鲁棒预测控制算法,将预测控制的待优化控制律考虑为线性反馈控制律,通过在线求解  $\min\text{-max}$  优化问题得到线性反馈增益。第三种为基于优化多项式反馈控制律的鲁棒预测控制算法,将预测控制的待优化控制律考虑为三次多项式反馈控制律,并以近似预测可达集与原点的距离为优化指标,通过在线优化求解三次多项式反馈控制的各项系数。

(4) 比较了用 CDF、UKF、EKF 为 PF 传递建议分布时各自的优缺点,指出 CDF 相较于 UKF 在速度上的优势及其相较于 EKF 在精度上的优势使 CDF 比 UKF、EKF 更适合为 PF 传递建议分布,并将中心差分粒子滤波器(CDPF)应用于鲁棒预测控制的状态估计。

(5) 针对非线性系统在跟踪问题中的预测控制器设计,将跟踪问题按期望输出的形式分为两类:对单个稳定期望输出点的逼近;对期望输出轨迹的

跟踪。针对这两种跟踪问题，分别设计了非线性系统的预测控制器。在第一种跟踪问题中，将系统对单个稳定期望输出点的逼近问题转换为对跟踪误差的镇定问题，得到了预测控制的稳定性条件。在第二种跟踪问题中，引入了大致跟踪的概念，指出了系统输出能大致跟踪期望输出所需满足的条件。针对一类非线性系统的第二种跟踪问题，提出了一种求取次优滚动控制律的方法。给出了一种基于线性近似的非线性模型预测控制方法，并在此基础上提出了一种基于 RBF 神经网络逼近的非线性模型预测控制方案。

(6) 介绍了航天器的末端自主交会技术，建立了基于视线坐标系的交会相对运动方程，并将预测控制应用于航天器的末端交会控制；针对飞行器的姿态机动问题，提出了一种将飞行器姿态驱动到设定姿态的模型预测控制算法。

## 8.2 存在的问题与下一步的研究方向

本书针对非线性预测控制做了一定的研究工作，但是所做工作只是针对非线性预测控制的某些问题进行讨论的。关于非线性预测控制，仍有大量的问题迫切需要进一步的研究与探讨，而我们目前的工作也存在着很多不足与缺陷，主要有以下几个方面：

(1) 设计了两种求取终端状态约束集的方法，即用外包集序列和子集序列逼近最大终端状态约束集。这两种方法得到的终端状态约束集都可能不是凸集，这会使得预测控制的在线优化不满足凸优化的条件，因此计算量会增大，如何在已经得到的终端状态约束集内求取一个最大的凸集需要进一步的研究。

(2) 在求取预测控制的终端代价函数时，我们只考虑了阶段指标函数为二次型的情况。当预测控制的阶段指标函数取其他形式时，如何求取终端代价函数是一个值得进一步研究的课题。

(3) 在求取鲁棒预测控制的终端代价函数与终端状态约束集时，我们所用的方法相对于系统不存在干扰时的方法来说比较独立，如何将系统不存在干扰时求取终端代价函数与终端状态约束集的方法拓展至干扰系统需要进一步的研究。

(4) 在设计鲁棒预测控制方法时，我们是在假设状态完全已知的情况下证明系统的稳定性的，而真正的状态则是需要用滤波方法来估计的，我们并没有证明用滤波估计值替代真实状态时系统的稳定性。因此，如何保证滤波估计状态收敛与控制系统稳定的统一性需要进一步的研究。

## 参考文献

- [1] Rechaler J, et al. Model Predictive Heuristic Control: Applications to Industrial processes [J]. Automatica, 1978, 14(5): 413-428.
- [2] Rouhani R, Mehra R K. Model Algorithmic Control (MAC): Basic Theoretical properties [J]. Automatica, 1982, 18(4): 401-414.
- [3] Cutler C R, Ramaker B L. Dynamic matrix control-a computer control algorithm [C]. Proceedings of the American control conference, San Francisco, 1980.
- [4] Garcia C E, Morari M. Internal model control, I. --A unifying review and some new results [J]. IEC Process Des. Dev, 1982, 21(2): 308-323.
- [5] Kuntze H B, Jacubasch A, Richalet J and Arber C. On the Predictive Functional Control of an Elastic Industrial Robot [C]. Proc. 25<sup>th</sup> CDC . Athens, Greece, 1986: 1877-1881.
- [6] Clarke D W, Monhtadl C and Tuffs P S. Generalize Predictive Control—Part I. The Basic Algorithm [J]. Automatica, 1987, 23(2):137-148.
- [7] Clarke D W, Monhtadl C and Tuffs P S. Generalize Predictive Control—Part II. Extensions and Interpretations [J]. Automatica, 1987, 23(2): 149-162.

- [8] Chen H and Allgower F. A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability [J]. *Automatica*, 1998, 34(10): 1205-1217.
- [9] Vu Trieu Minh and Nitin Afzulpurkar. A comparative study on computational schemes for nonlinear model predictive control [J]. *Asian journal of control*, 2006, 8(4): 324-331.
- [10] Kwon W H, Pearson A E. On feedback stabilization of time-varying discrete linear systems [J]. *IEEE Transactions on Automatic Control*, 1978, 23(3): 479-481.
- [11] Scokaert P O M, Clarke D W. Stabilising properties of con-strained predictive control [J]. *IEE Proceedings, Control Theory and Applications*, 1994, 141(5): 295-304.
- [12] Clarke D W, Scattolini R. Constrained receding-horizon predictive control [J]. *IEE Proceedings, Control Theory and Applications*, 1991, 138(4): 347-354.
- [13] Keerthi S S, Gilbert E G. Optimal infinite-horizon feedback laws for a general class of constrained discrete-time sys-tems: stability and moving-horizon approximations [J]. *Journal of Optimization Theory and Applications*, 1988, 57(2):265-293.
- [14] Mayne D Q, Hannah Michalska. Receding horizon control of nonlinear systems [J]. *IEEE transactions on automatic control*, 1990, 35(7): 814-824.
- [15] Michalska H, Mayne D Q. Robust receding horizon control of constrained nonlinear systems [J]. *IEEE transactions on automatic control*, 1993, 38(11): 1623-1633.
- [16] Limon D, Alamo T, Salas F, Camacho E F. On the stability of constrained

- MPC without terminal constraint [J]. IEEE transactions on automatic control, 2006, 51(5): 832-836.
- [17] Ali Jadbabaie, John Hauser. On the stability of receding horizon control with a general terminal cost [J]. IEEE transactions on automatic control, 2005, 50(5): 674-678.
- [18] Chmielewski D. and Manousiouthakis V. On constrained infinite-time linear quadratic optimal control [J]. Systems & Control Letters, 1996, 29(3): 121-129.
- [19] De Nicolao G, Magni L and Scattolini R. Stabilizing nonlinear receding horizon control via a nonquadratic penalty [C]. Proceedings of the IMACS multiconference CESA, Lille, France, 1996: 185-187.
- [20] Gilbert E G. and Tan K T. Linear systems with state and control constraints: the theory and application of maximal output admissible sets [J]. IEEE Transactions on Automatic Control, 1991, AC-36: 1008-1020.
- [21] Parisini T. and Zoppoli R. A receding horizon regulator for nonlinear systems and a neural approximation [J]. Automatica, 1995, 31(10): 1443-1451.
- [22] Sznaier M. and Damberg M J. Suboptimal control of linear systems with state and control inequality constraints [C]. Proceedings of the 26th IEEE conference on decision and control, Los Angeles, 1987: 761-762.
- [23] Scokaert P O M. and Rawlings J B. Constrained linear quadratic regulation [J]. IEEE Transactions on Automatic Control, 1998, 43(8), 1163-1169.
- [24] Simone Loureiro de Oliveira Kothare and Manfred Morari, Contractive model predictive control for constrained nonlinear systems [J]. IEEE transactions on automatic control, 2000, 45(6): 1053-1071.

- [25] Mayne D Q, Rawlings J B, Rao C V, Scokaert P O M. Constrained model predictive control: stability and optimality [J]. *Automatica*, 2000, 36(6): 789-814.
- [26] De Nicolao G, Magni L and Scottolini R. Stabilizing receding-horizon control of nonlinear time-varying systems [J]. *IEEE Transactions on Automatic Control*, 1998, 43(7): 1030-1036.
- [27] Magni L, et al. A stabilizing model-based predictive control algorithm for nonlinear systems [J]. *Automatica*, 2001, 37(9): 1351-1362.
- [28] Lee J W, Kwon W H and Choe J. On stability of constrained receding horizon control with finite terminal weighting matrix [J]. *Automatica*, 1998, 34(12): 1607-1612.
- [29] Lee J W. Exponential stability of constrained receding horizon control with terminal ellipsoid constraints [J]. *IEEE Transactions on Automatic Control*, 2000, 45(1): 83-88.
- [30] Scokaert P O M., Rawlings J B. and Meadows E S. Discrete-time stability with perturbations: application to model predictive control [J]. *Automatica*, 1997, 33(3): 463-470.
- [31] De Dona J A, et al. Enlarged terminal sets guaranteeing stability of receding horizon control [J]. *Systems & Control Letters*, 2002, 47(1): 57-63.
- [32] Limon D, Alamo T and Camacho E F. Enlarging the domain of attraction of MPC controllers [J]. *Automatica*, 2005, 41(4): 629-635.
- [33] Yang T H and Polak E. Moving horizon control of nonlinear systems with input saturation, disturbances and plant uncertainty [J]. *International Journal of control*, 1993, 58(4): 875-903.

- [34] De Oliveira S L, Morari M. Robust model predictive control for nonlinear systems [C]. Proceedings of the 33th IEEE conference on Decision and Control, 1994, 4: 3561-3566.
- [35] Alejandro H. González and Darci Odloak. Enlarging the domain of attraction of stable MPC controllers, maintaining the output performance [J]. Automatica, 2009, 45(4): 1080-1085.
- [36] Cannon M, Deshmukh V and Kouvaritakis B. Nonlinear model predictive control with polytopic invariant sets[J]. Automatica, 2003, 39(8): 1487-1494.
- [37] Ong C J, Sui D, Gilbert E G. Enlarging the terminal region of nonlinear model predictive control using the support vector machine method [J]. Automatica, 2006, 42(6): 1011-1016.
- [38] Limon D, et al. On the stability of constrained MPC without terminal constraint [J]. IEEE transactions on automatic control, 2006, 51(6): 832-836.
- [39] Lars Grüne and Anders Rantzer. On the Infinite Horizon Performance of Receding Horizon Controllers [J]. IEEE transactions on automatic control, 2008, 53(9): 2100-2111.
- [40] Bei Hu and Arno Linnemann. Toward Infinite-Horizon Optimality in Nonlinear Model Predictive Control [J]. IEEE transactions on automatic control, 2002, 47(4): 679-682.
- [41] 耿晓军, 席裕庚. 基于 HM 非线性模型的滚动时域  $H_\infty$  控制[J]. 自动化学报, 2000, 26 (1): 68-73.
- [42] Kim K B. Disturbance attenuation for constrained discrete-time systems via receding horizon controls [J]. IEEE Transactions on Automatic Control, 2004, 49(5): 797-801.



- [43] Kothare M V, Balakrishnan V, Morari M. Robust constrained model predictive control using linear matrix inequalities [J]. Automatica, 1996, 32(10): 1361-1379.
- [44] Lee Y I, Kouvaritakis B. Robust receding horizon predictive control for systems with uncertain dynamics and input saturation [J]. Automatica, 2000, 36(10): 1497-1504.
- [45] Limon D, Bravo J M, Alamo T and Camacho E F. Robust MPC of constrained nonlinear systems based on interval arithmetic [J]. IEE Proc.Control Theory Applications, 2005, 152(3): 325-332.
- [46] Bravo J M, Alamo T, Camacho E F. Robust MPC of constrained discrete-time nonlinear systems based on approximated reachable sets [J]. Automatica, 2006, 42(10): 1745-1751.
- [47] Limon D, Alamo T, Salas F and Camacho E F. Input to state stability of min-max MPC controllers for nonlinear systems with bounded uncertainties [J]. Automatica, 2006, 42(5): 797-803.
- [48] 何德峰, 季海波, 郑涛. 持续有界扰动下的非线性  $H_\infty$  鲁棒预测控制[J]. 自动化学报, 2008, 34 (2): 215-219.
- [49] 耿晓军, 席裕庚. 不确定系统的滚动时域  $H_\infty$  控制设计[J]. 控制与决策, 2000, 15 (2): 149-152.
- [50] Casavola A, Famularo D and Franz G. Robust constrained predictive control of uncertain norm-bounded linear systems [J]. Automatica, 2004, 40(11): 1865-1876.
- [51] Angeli D, Casavola A and Mosca E. Ellipsoidal low-demanding MPC schemes for uncertain polytopic discrete-time systems [C]. In: Proceedings

- of the 41th IEEE Conference on Decision and Control. Las Vegas, USA: IEEE, 2002. 2935-2939.
- [52] Manye D Q, Seron M M and Rakovic S V. Robust model predictive control of constrained linear systems with bounded disturbances [J]. Automatica, 2005, 41(2): 219-224.
- [53] Mayne D Q, et al. Robust output feedback model predictive control of constrained linear systems [J]. Automatica, 2006, 42(7): 1217-1222.
- [54] Rakovi S V, et al. Optimized robust control invariance for linear discrete-time systems: theoretical foundations [J]. Automatica, 2007, 43(5): 831-841.
- [55] Fukushima H and Bitmead R R. Robust constrained predictive control using comparison model [J]. Automatica, 2005, 41(1): 97-106.
- [56] Scokaert P Q M and Mayne D Q. Min-max feedback model predictive control for constrained linear systems [J]. IEEE Transactions on Automatic Control, 1998, 43(8): 1136-1142.
- [57] Chisci L, Rossiter J A and Zappa G. Systems with persistent disturbances: predictive control with restricted constraints [J]. Automatica, 2001, 37(7): 1019-1028.
- [58] Mosca E. Predictive switching supervisory control of persistently disturbed input-saturated plants [J]. Automatica, 2005, 41(1): 55-67.
- [59] Gelb A, et al. Applied optimal estimation [M], MIT Press, Cambridge, MA, 1974.
- [60] Anderson B D. and Moore J B. Optimal Filtering[M]. Prentice-Hall, New Jersey, 1979.

- [61] Julier S J and Uhlmann J K. A new extension of the Kalman filter to nonlinear systems [C]. Proc. SPIE, Signal Processing, Sensor Fusion and Target Recognition VI, 1997, 3068:182-193.
- [62] Wan E A and van der Merwe R. The Unscented Kalman Filter for nonlinear estimation [C]. Proceedings of Symposium 2000 on Adaptive Systems for Signal Processing, Communication and Control (AS-SPCC), IEEE, Lake Louise, Alberta, Canada, 2000, 153-158.
- [63] Schei T S. A finite-difference method for linearization in nonlinear estimation algorithms [J]. Automatica, 1997, 33(11): 2051-2058.
- [64] Magnus Norgaard, Niels K Poulsen and Ole Ravn. New developments in state estimation for nonlinear systems [J]. Automatica, 2000, 36(11): 1627-1638.
- [65] Magnus Norgaard, Niels K Poulsen and Ole Ravn. Advances in Derivative-Free State Estimation for Nonlinear Systems, Technical report: IMM-REP-1998-15 (revised edition), Technical University of Denmark, Denmark, April 7, 2000.
- [66] Brian G Saulson and Kuo-Chu Chang. Nonlinear estimation comparison for ballistic missile tracking [J]. Society of Photo-Optical Instrumentation Engineers, 2004, 43(6): 1424-1438.
- [67] Doucet A. On sequential simulation-based methods for Bayesian filtering, Technical Report CUED/F-INFENG/TR 310, Department of Engineering, Cambridge University, 1998.
- [68] Sanjeev Arulampalam M, et al, A Tutorial on Particle Filters for Online Nonlinear/ Non-Gaussian Bayesian Tracking [J]. IEEE Transactions on signal processing, 2002, 50(2): 174 -188.

- [69] Doucet A, Gordon N J and Krishnamurthy V. Particle filters for state estimation of jump Markov linear systems, Technical Report CUED/F-INFENG/TR 359, Cambridge University Engineering Department. 1999.
- [70] Liu J S and Chen R. Sequential Monte Carlo methods for dynamic systems [J]. Journal of the American Statistical Association, 1998, 93(443): 1032-1044.
- [71] Andrieu C, de Freitas J F G and Doucet A. Sequential MCMC for Bayesian model selection [C]. IEEE Higher Order Statistics Workshop, Ceasarea, Israel, 1999, 130-134.
- [72] de Freitas J F G, et al. Sequential Monte Carlo methods to train neural network models [J]. Neural Computation, 2000, 12(4): 955-993.
- [73] Rudolph van der Merwe, et al, The Unscented Particle Filter, Technical Report CUED/F-INFENG/TR 380, August 16, 2000.
- [74] 石勇, 韩崇昭. 二阶中心差分粒子滤波算法[J]. 西安交通大学学报, 2008, 42 (4): 409-413.
- [75] Smita Sadhu, et al. Central Difference Formulation of Risk-Sensitive Filter [J]. IEEE Signal processing letters, 2007, 14(6): 421-424.
- [76] Schweppe F. Recursive state estimation: unknown but bounded errors and system inputs [J]. IEEE Transactions on Automatic Control, 1968, 13(1): 22-28.
- [77] Maksarov D G and Norton J P. State bounding with ellipsoidal set description of uncertainty [J]. International Journal of Control, 1996, 65(5): 847-866.
- [78] Kieffer M, Jaulin L and Walter E. Guaranteed recursive non-linear state

- estimation using interval analysis [J]. International Journal of Adaptive Control and Signal Processing, 2002, 16(3): 193-218.
- [79] Jaulin L. Nonlinear bounded-error state estimation of continuous-time systems [J]. Automatica, 2002, 38(6): 1079-1082.
- [80] Scholte E, Campbell M. A nonlinear set-membership filter for on-line applications [J]. International Journal of Robust and Nonlinear Control, 2003, 13(10): 1337-1358.
- [81] Maksarov D G and Norton J P. Computational efficient algorithms for state estimation with ellipsoidal approximations [J]. International Journal of Adaptive Control and Signal Processing, 2002, 16(5): 411-434.
- [82] Crassidis J L and Markley F L. Predictive filtering for nonlinear system [J]. Journal of Guidance, Control and Dynamics, 1997, 20(3): 566-572.
- [83] Crassidis J L and Markley F L. Predictive filtering for attitude estimation without rate sensors [J]. Journal of Guidance, Control and Dynamics, 1997, 20(3): 522-527.
- [84] Lin Yurong, Deng Zhenglong Shao Changsheng. Nonlinear predictive filter for satellite attitude estimation using star sensors [C]. Proceedings of the 3rd World Congress on Intelligent Control and Automation, Hefei, P.R. China, 2000, 2949-2953.
- [85] Bemporad A, Casavola A and Mosca E. Nonlinear control of constrained linear systems via predictive reference management [J]. IEEE Transactions on Automatic Control, 1997, 42(3): 340-349.
- [86] Blanchini F and Miani S. Any domain of attraction for a linear constrained

- system is a tracking domain of attraction [J]. SIAM Journal on Control and Optimization, 2000, 38: 971-994.
- [87] Fiacchini M, et al. Predictive control of a linear motor for tracking of constant references [C]. In Proceedings of the CDC. 2006.
- [88] Chisci L and Zappa G. Dual mode predictive tracking of piecewise constant references for constrained linear systems [J]. International Journal of Control, 2003, 76(1): 61-72.
- [89] Limon D, et al. MPC for tracking piecewise constant references for constrained linear systems [J]. Automatica, 2008, 44: 1-6.
- [90] Alvarado I. Model predictive control for tracking constrained linear systems [D]. Ph.D. thesis. Universidad de Sevilla, 2007.
- [91] Eelco Scholte and Mark E. Campbell. Robust nonlinear model predictive control with partial state information [J]. IEEE Transactions on Control Systems Technology, 2008, 16(4): 636-651.
- [92] Haralambos Sarimveis, George Bafas. Fuzzy model predictive control of nonlinear processes using genetic algorithms [J]. Fuzzy Sets and Systems, 2003, 139(1): 59-80.
- [93] Daniel V D, YU Tang. Adaptive robust fuzzy control of nonlinear systems [J]. IEEE Transactions on Systems, Man, and Cybernetics, 2004, 34(3): 1596-1601.
- [94] 邹涛, 王昕, 李少远. 基于混合逻辑的非线性系统多模型预测控制[J]. 自动化学报, 2007, 33 (2): 188-192.
- [95] 王蓬, 李少远. 一类非线性系统的多模型预测控制[J]. 控制与决策, 2007, 22 (10): 1114-1118.

- [96] 张日东, 王树青, 李平. 基于支持向量机的预测控制[J]. 自动化学报, 2007, 33 (10): 1066-1073.
- [97] 罗秋滨, 岳中哲, 关立雁. 多项式逼近建模的非线性系统预测控制[J]. 电机与控制学报, 2008, 12 (1): 84-88.
- [98] 沈永良, 胡致强, 赵建华. 基于动态非线性逼近的非线性系统预测控制[J]. 控制与决策, 2007, 22 (1): 100-104.
- [99] 沈继红. 灰色系统理论预测方法研究及其在舰船运动预报中的应用[D]. 哈尔滨工程大学博士学位论文, 2002.
- [100] 郭健, 陈庆伟, 朱瑞军. 一类非线性系统的自适应预测控制[J]. 控制理论与应用, 2002, 19 (1): 68-72.
- [101] 王洪瑞, 陈志旺, 李建雄. 非线性系统参数自适应直接广义预测控制[J]. 自动化学报, 2007, 33 (10): 1110-1114.
- [102] Martin Rau, Dierk Schroder, Model Predictive Control with Nonlinear State Space Models [C]. IEEE, AMC 2002-Maribor, Slovenia, 2002, 136-141.
- [103] Bie T D, et al. Kernel-based data fusion for gene prioritization [J]. Bioinformatics, 2007, 23(13): 125-132.
- [104] Bi J B and Zhang T. Support Vector Classification with Input Data Uncertainty [C]. In: Saul L K, Weiss Y and Bottou L, eds. Advances in Neural Information Processing Systems, the Eighteenth Annual Conference on Neural Information Processing Systems, Vancouver, 2004. Massachusetts: The MIT Press, 2004, 17: 161-168.
- [105] Boser B E, Guyon I M and Vapnik V N. A Training Algorithm for Optimal Margin Classifiers [C]. Proceedings of the Fifth Annual ACM Conference

- on Computational Learning Theory, Pittsburgh, PA, USA:ACM, 1992, 144-152.
- [106] Cortes C and Vapnik V N. Support Vector Networks [J]. Machine Learning, 1995, 20(3): 273-297.
- [107] Vapnik V N. The Nature of Statistical Learning Theory [M]. New York: Springer-Verlag, 1995.
- [108] Vapnik V N. Statistical Learning Theory [M].Wiley, 1998.
- [109] Osuna E, Freund R and Guosi F. Training Support Vector Machines: An Application to Face Detection [C]. In: Proceedings of IEEE Conference on CVPRE, Puerto Rico, 1997, 130-136.
- [110] Joachims T. Text Categorization with Support Vector Machines: Learning With Many Relevant Features [J]. Lecture Notes in Computer Science, 1998, 1398: 137-142.
- [111] Chapelle O, Haffner P and Vapnik V N. SVMs for Histogram-Based Image Classification [J]. IEEE transactions on Neural Networks, 1999, 10(5): 1055-1065.
- [112] Moore E. Interval analysis [M]. Prentice Hall, 1996.
- [113] Kearfott R B. Rigorous global search: continuous problems [M]. Kluwer, 1996.
- [114] Hansen E. Global optimization using interval analysis [M]. Marcel Dekker, 1992.
- [115] Carter T. Fuel-optimal rendezvous near a point in general keplerian orbit [J]. Journal of Guidance Control and Dynamics, 1987, 10(6): 567-573.



- [116] 于绍华, 杨林娜. 空间交会的末端控制模式[J]. 宇航学报, 1998, 19 (3): 16-22.
- [117] 陈统, 徐世杰. 非合作式自主交会对接的终端接近模糊控制[J]. 宇航学报, 2006, 27 (3): 416-421.
- [118] Fagiano L, Teel A R. On generalized terminal state constraints for model predictive control [J]. 2012, arXiv: 1207.0788v1.
- [119] 胡跃明, 胡终须, 毛宗源, 等. 非线性控制系统的近似化方法[J]. 控制理论与应用, 2001, 18 (2): 160-165.
- [120] 戴文战, 娄海川, 杨爱萍. 非线性系统神经网络预测控制研究进展[J]. 控制理论与应用, 2009, 26 (5): 521-530.
- [121] Kwaku O T, Phillip D S, Thomas J M. Model predictive control of an industrial packed bed reactor using neural networks [J]. Journal of Process Control, 1995, 5(1): 19-27.
- [122] Kodogiannis V S, Lisboa J G, Lucas J. Neural network modeling and control for underwater vehicles [J]. Artificial intelligence in Engineering, 1996, 10(3): 203-212.
- [123] Trajanoski Z, Regittinig W, Wach P. Neural predictive controller for closed-loop control of glucose using the subcutaneous route: a simulation study [J]. Control Engineering Practice, 1997, 5(12): 1727-1730.
- [124] Andre M S, Francis J D. Multivariable nonlinear control applications for a high purity distillation column using a recurrent dynamic neuron model [J]. Journal of Process Control, 1997, 7(4): 255-268.
- [125] Benne M, Perez B G, Herve P. Artificial neural networks for modeling and

- predictive control of an industrial evaporation process [J]. Journal of Food Engineering, 2000, 46(4): 227-234.
- [126] Gao F R, Wang F L, Li M Z. A simple nonlinear controller with diagonal recurrent neural network [J]. Chemical Engineering Science, 2000, 55(7): 1283-1288.
- [127] Rasit K. Design and performance of an intelligent predictive controller for a six-degree-of-freedom robot using the Elman network [J]. Information Science, 2006, 176(12): 1781-1799.
- [128] Hui P, Jun W, Garba I, Qiu L D, Kazushi N. Nonlinear system modeling and predictive control using the RBF nets-based quasi-linear ARX model [J]. Control Engineering Practice, 2009, 17: 59-66.
- [129] Maciej L. Computationally Efficient Nonlinear Predictive Control Based on RBF Neural Multi-models [C]. Proc of the 9th international conference on Adaptive and natural computing algorithms. Berlin: Springer Verlag, 2009, 89-98.
- [130] 李会军, 肖兵. 一种无约束多步递归神经网络预测控制器[J]. 控制理论与应用, 2012, 29 (5): 642-648.
- [131] 刘贺平, 张兰玲, 孙一康. 基于多层局部回归神经网络的多变量非线性系统预测控制[J]. 控制理论与应用, 2001, 18 (2): 298-300.
- [132] 王亚锋, 孙富春, 张友安, 等. 一种跟踪问题中的次优非线性预测控制算法[J]. 控制与决策, 2009, 24 (11): 1682-1686.
- [133] Magnus N, Niels K P, Ole R. Advances in Derivative-Free State Estimation for Nonlinear Systems [R], Technical report: IMM-REP-1998-15 (revised edition), Technical University of Denmark, Denmark, April 7, 2000.

- [134] Schei T S. A finite difference method for linearization in nonlinear estimation algorithms [J]. Automatica, 1997, 33(11): 2051-2058.
- [135] Magnus N, Niels K P, Ole R. New developments in state estimation for nonlinear systems [J]. Automatica, 2000, 36(11): 1627-1638.
- [136] 徐丽娜. 神经网络控制 (第三版) [M]. 北京: 电子工业出版社, 2009.
- [137] 于占东, 王庆超. 飞行器姿态的一种鲁棒自适应模糊解耦控制[J]. 宇航学报, 2003, 24 (4): 368-373.
- [138] Chang-Hee Won. Comparative study of various control methods for attitude control of LEO satellite [J]. Aerospace Science and Technology. 1999, No.5:323-333.
- [139] Huang J, Lin C F. Application of sliding mode control to Bank-to Turn missile system [C]. Aerospace Control Systems, The First IEEE Regional Conference. 1993: 569-573.
- [140] Keviczky T, Balas G J. Receding horizon control of an F-16 aircraft: a comparative study [J]. Control Engineering Practice, 2006, 14(9): 1023-1033.
- [141] Silani E, Lovera M. Magnetic spacecraft attitude control: a survey and some new results [J]. Control Engineering Practice, 2005, 13(3): 357-371.
- [142] 孙光, 霍伟. 卫星姿态直接自适应模糊预测控制[J]. 自动化学报, 2010, 36 (8): 1151-1159.
- [143] Alexis K, Nikolakopoulos G, Tzes A. Switching model predictive attitude control for a quadrotor helicopter subject to atmospheric disturbances [J]. Control Engineering Practice, 2011, 19(10): 1195-1207.

- [144] Gavilan F, Vazquez R, Camacho E F. Chance-constrained model predictive control for spacecraft rendezvous with disturbance estimation [J]. Control Engineering Practice, 2012, 20(2): 111-122.
- [145] 吴森堂, 费玉华. 飞行控制系统[M]. 北京: 北京航空航天大学出版社, 2005.
- [146] 谢岚. 高空长航时无人机飞行控制系统设计[D]. 湖南大学, 2011.
- [147] 韩艳铎, 周凤岐, 周军. 基于反馈线性化和变结构控制的飞行器姿态控制系统设计[J]. 宇航学报, 2004, 25 (6): 637-641.